QikProp

2.0 User Manual—Unix

QikProp

2.0 User Manual—Unix

Copyright © 2002 Schrödinger, L.L.C. All rights reserved.
Schrödinger, QikProp, and Maestro are trademarks of Schrödinger, L.L.C.
MacroModel is a registered trademark of Schrödinger, L.L.C.

August 2002
RV2000201

Table of Contents

Chapter 1: User Manual Overview 1

1.1 Purpose 1

1.2 Document Conventions 1

Chapter 2: The QikProp Product Family 3

2.1 QikProp 3

2.2 QikFit 6

2.3 QikSim 6

Chapter 3: The Maestro User Interface 7

3.1 The Maestro Main Application Window 7

3.2 The Project Facility 8

3.3 Maestro Workflow Paradigm 10

3.4 Picking Atoms 10

3.4.1 Pick States 10

3.4.2 Picking Mode 11

3.5 Mouse Functions 11

3.5.1 Specialized Mouse Functions in the Project Facility 12

3.6 Sidebar Controls 13

3.7 Maestro Command Scripts 14

Chapter 4: Launching the Programs 15

4.1 Introduction 15

4.2 Environment Variables 15

4.3 Launching QikProp from Maestro 16

4.4 Launching Command Line QikProp 16

4.5 Launching QikFit 17

4.6 Launching QikSim 17

4.7 Displaying Command Syntax 17

Chapter 5: QuickTopics 19
5.1 Introduction 19
5.2 Getting Ready to Use QikProp from Maestro 19
Launching Maestro 19
Specifying a Maestro Working Directory 20
Creating a Working Tutorial Directory 21
5.3 QuickTopic Exercises 22
Importing Structures to Use as QikProp Input 22
Using QikProp to Generate a Training Set 24
Running QikProp in Fast Processing Mode 26
Sorting Project Table Entries 27
Creating a Training File for use with QikFit 29
Using QikProp to Generate a Test File 30
Running QikFit 31
Predicting New Properties with the QPmyfits File 33
Creating a Plot of Project Table Properties 34
Creating A Scatter Plot 36
Running QikSim and Examining Output Data 38
5.4 Summary 39

Chapter 6: Maestro QikProp 41
6.1 Opening the QikProp Panel 41
6.2 Naming the QikProp Job 41
6.3 Selecting a Host Computer 41
6.4 Using the QPlimits File 43
6.5 Specifying an Input Source 43
6.6 Selecting a Job Output Incorporation Scheme 44
6.7 Starting a QikProp Job 44
6.8 Monitoring the Job 45

Chapter 7: QikProp Output 47
7.1 Output File Types 47
7.1.1 The <jobname>-out.mae File 47
7.1.2 The <jobname>.mae File 47

7.1.3 The <jobname>.qpsa File 48
7.1.4 The <jobname>.out File 48
7.1.5 The <jobname>.CSV File 48
7.1.6 The <jobname>.warning File 49
7.2 Output File Directory 49

Chapter 8: Command Line QikProp 51

8.1 Introduction 51
8.2 Issuing the 'qikprop' Command 51
8.3 Input Files 52

Chapter 9: QikFit 55

9.1 Introduction 55
9.2 Running QikFit 56
9.3 The Training File 57
9.4 The Test File 58
9.5 Significance Cutoff Values 58
9.6 Running on a Remote Host 59
9.7 Output Files 59
9.7.1 The <jobname>.out File 59
9.7.2 The <jobname>.myfits File 59
9.8 Using QikFit Output To Predict New Properties 60

Chapter 10: QikSim 61

10.1 Introduction 61
10.2 Command Syntax 61
10.3 Input 62

Chapter 11: Bibliography 65

11.1 QikProp 65
11.2 QikFit 65

Chapter 12: Help 67

12.1 Getting Help on Program Installation 67

12.2 Getting Help from within Maestro 67
12.3 QikProp Help from the Command Line 68
12.4 Technical Support 68

Appendix A: Configuring Job Control 69
A.1 Introduction 69
A.2 The Location of the Scratch Directory 69
A.3 Preparing for Remote Job Submission 70
A.4 The schrodinger.hosts File 71
A.4.1 Structure of the schrodinger.hosts File 71
A.4.2 Selection of the schrodinger.hosts File 73
A.4.3 Testing the schrodinger.hosts File 73

Appendix B: Command Line Job Control 75
B.1 Introduction 75
B.2 Jobcontrol Syntax and Supported Commands 75
B.3 Using jobcontrol to Purge the Job Database 76
B.4 Default Behaviors 76
B.5 Remote Job Submission 77
B.6 The Job Control Scratch Directory 77
B.7 The Job Record 78

Appendix C: Sample Output Files 81
C.1 Sample Regression Output and Commentary 81
C.2 Trend Vector Output 85
C.3 Sample myfits File 88
C.4 QikSim Output 90

Appendix D: Sample Input Files 93
D.1 Sample mol File 93
D.2 Sample Maestro-formatted File 95

Index 99

Chapter 1: User Manual Overview

1.1 Purpose

This manual contains an introduction to the Maestro™ graphical user interface (GUI) and in-depth instruction on using the QikProp™ adsorption, distribution, metabolism, and excretion (ADME) prediction software program.

For information about features that are new in this release, see the *QikProp 2.0 Release Notes*. For more information on using Maestro, see the Maestro online help or the *Maestro User Manual*. For technical information on the accuracy of QikProp, see the *QikProp 2.0 Technical Information* document.

1.2 Document Conventions

Throughout this manual, reference is made to Maestro GUI components such as panels, buttons, toggle buttons, radio buttons, Sidebar options, menus, and menu options. GUI components whose labels appear in the GUI are set in Helvetica. Elements that are not labeled in the GUI are capitalized but not set in Helvetica.

In addition to GUI components, the names of keyboard keys are set in Helvetica. Machine input and output are set in `Courier`. File names, command script names, and entry names are set in *italics*. Directory names and project names are set in **bold**.

The QuickTopics chapter of this manual contains exercise sets that are intended to provide fast, streamlined instruction for performing common QikProp tasks. Each of these QuickTopic exercise sets begins with the icon:

In this document, to *type* a command means to type the required text in the specified location, and to *enter* a command means to type the required text and then press the Enter key.

The environment variable SCHRODINGER is used for the directory in which your Schrödinger software is installed.

Chapter 2: The QikProp Product Family

2.1 QikProp

QikProp is a quick, accurate, easy-to-use adsorption, distribution, metabolism, and excretion (ADME) prediction program designed by Professor William L. Jorgensen. Based on years of Jorgensen's research, QikProp predicts physically significant descriptors and pharmaceutically relevant properties for organic structures, either individually or in batches.

In addition to predicting molecular properties, QikProp provides ranges for comparing a particular molecule's properties with those of 95% of known drugs. QikProp also flags 30 types of reactive functional groups that may cause false positives in high-throughput screening (HTS) assays.

QikProp has two available processing modes. By default, QikProp runs in normal mode, predicting 36 properties for as many as 10,000 molecules in an hour. Alternatively, QikProp can be run in fast processing mode, which predicts 32 properties rather than the default 36. In fast mode, QikProp is able to evaluate approximately 150,000 compounds per hour on a 1 GHz Pentium processor. The *QPlimits* file must be edited to switch from normal to fast processing mode. This file can also be edited to specify the range of values which cause a molecule to be flagged as dissimilar to other known drugs. See section 6.4 "Using the QPlimits File" on page 43 for more details about the *QPlimits* file.

QikProp output can be used as input for the QikFit™ and QikSim™ modules. The QikFit module performs linear regression analyses for experimentally determined molecular properties. The resulting regression equations can then be integrated back into QikProp and used to predict the experimental property for additional, structurally similar molecules. QikSim performs similarity/diversity analyses on lists of structures based on QikProp descriptors.

QikProp can be run either from the Maestro GUI or from the command line.

For each successfully processed molecule, QikProp produces the following descriptors and properties. Those that are not predicted in fast mode are marked with a dagger (†).

molecule name	Molecule name taken from the title line in the input structure file. If the title line is blank, the input file name is used.
#stars	Number of property or descriptor values that fall outside the 95% range of similar values for known drugs. Outlying descriptors and predicted properties are denoted with asterisks (*) in the *.qpsa* file. A large number of stars suggests that a molecule is less drug-like than molecules with few stars.
#amine	Number of non-conjugated amine groups.
#nitro	Number of nitro groups.
#acid	Number of carboxylic acid groups.
#amide	Number of non-conjugated amide groups.
#rotor	Number of non-trivial (not CX3), non-hindered (not alkene, amide, small ring) rotatable bonds.
#rctvFG	Number of reactive functional groups; the specific groups are listed in the <*jobname*>*.out* file. The presence of these groups can lead to false positives in HTS assays and to decomposition/reactivity/ toxicity problems *in vivo*.
CNS	Predicted central nervous system activity on a -2 (inactive) to +2 (active) scale.
MW	Molecular weight of the molecule.
dipole†	Computed dipole moment of the molecule.
SASA	Total solvent accessible surface area (SASA) in square ångströms using a probe with a 1.4Å radius.
FOSA	Hydrophobic component of the SASA (saturated carbon and attached hydrogen).
FISA	Hydrophilic component of the SASA (SASA on N, O, and H on heteroatoms).
PISA	π (carbon and attached hydrogen) component of the SASA.
WPSA	Weakly polar component of the SASA (halogens, P, and S).
volume	Total solvent-accessible volume in cubic ångströms using a probe with a 1.4Å radius.

donorHB	Estimated number of hydrogen bonds that would be donated by the solute to water molecules in an aqueous solution.
accptHB	Estimated number of hydrogen bonds that would be accepted by the solute from water molecules in an aqueous solution.
dip^2/V†	Square of the dipole moment divided by the molecular volume. This is the key term in the Kirkwood-Onsager equation for the free energy of solvation of a dipole with volume V.
ACxDN^.5/SA	Index of cohesive interaction in solids. This term represents the relationship $(accptHB(\sqrt{donorHB}))/(SA)$; see *Bioorg. Med. Chem. Lett.* 10, 1155 (2000).
glob	Globularity descriptor, $(4\pi r^2)/(SASA)$, where *r* is the radius of a sphere with a volume equal to the molecular volume. Globularity is 1.0 for a spherical molecule.
QPpolrz	Predicted polarizability in cubic ångströms.
QPlogPC16	Free energy of solvation in hexadecane.
QPlogPoct	Free energy of solvation in octanol.
QPlogPw	Free energy of solvation in water.
QPlogPo/w	Predicted octanol/water partition coefficient.
QPlogS	Predicted aqueous solubility, log S. S in moles/liter is the concentration of the solute in a saturated solution that is in equilibrium with the crystalline solid.
BIPCaco	Predicted apparent Caco-2 cell permeability in nm/sec using the Boehringer-Ingelheim scale.
AffyPCaco	Predicted apparent Caco-2 cell permeability in nm/sec using the Affymax scale.
QPlogBB	Predicted brain/blood partition coefficient.
AffyPMDCK	Predicted apparent MDCK cell permeability in nm/sec using the Affymax scale.
QPlogKp	Predicted skin permeability.
IP(ev)†	PM3 calculated ionization potential.
EA(eV)†	PM3 calculated electron affinity.
#metabol	The number of likely metabolic reactions.
QPLog Khsa	Prediction of binding to human serum albumin.

Additional information on many of these properties can be found in the directory **$SCHRODINGER/qikprop-v<version>/docs**, where <version> is the five digit version number of your QikProp installation.

2.2 QikFit

QikFit generates optimized regression equations for the correlation of experimental data with sets of descriptors, such as those generated by QikProp. QikFit output can then be used by QikProp to predict the experimental property for any number of additional, structurally similar molecules. Alternatively, QikFit can be used independently from QikProp.

In addition to regression analysis, QikFit can perform trend vector analyses. Trend vector analysis is generally used when there are a large number of descriptors and the ratio of molecules to descriptors is not high. For regression analyses, this ratio should be at least five.

There is currently no Maestro interface for QikFit. This module must be run from the command line.

2.3 QikSim

QikSim performs similarity/diversity analyses on structures in the input file based on Euclidean distances and Tanimoto coefficients. All included structures are ranked in comparison to a user-specified probe molecule. Output from QikProp can be used as input for QikSim.

There is currently no Maestro interface for QikSim. This module must be run from the command line.

Chapter 3: The Maestro User Interface

3.1 The Maestro Main Application Window

When you start Maestro, the Main Application Window is displayed (Figure 3.1). This window is composed of the Title bar, the Auto-Help window, the Main MenuBar, the Sidebar, the Workspace, the Command Input Area, and the Top View window.

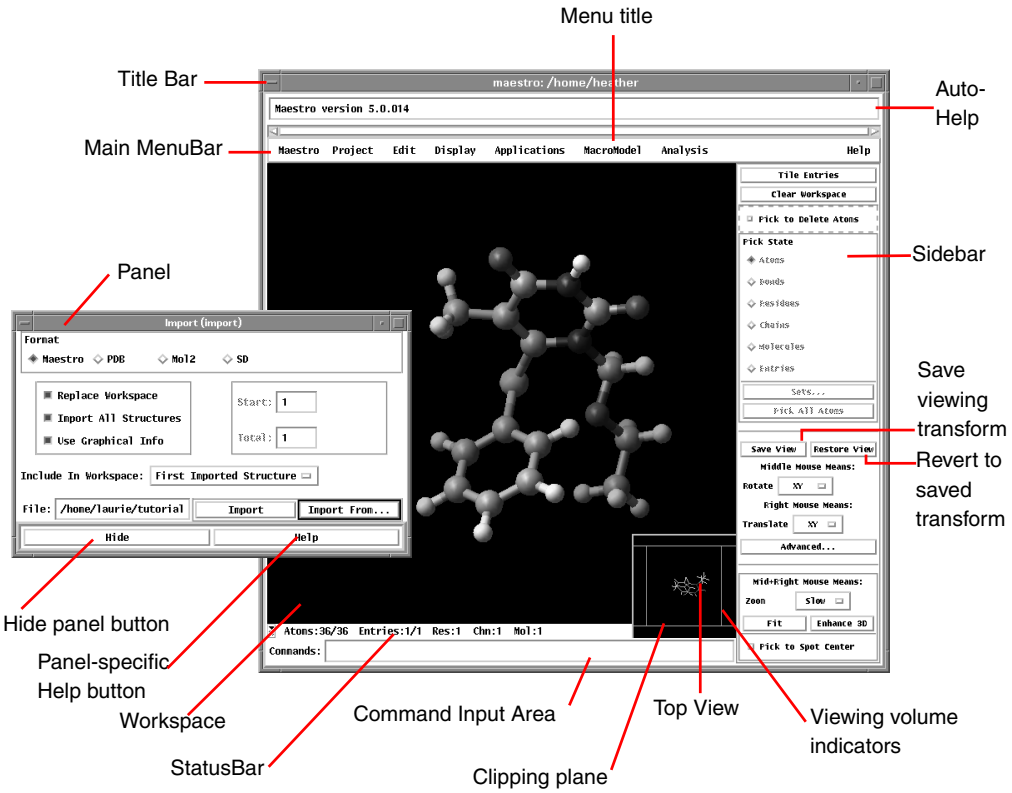

Figure 3.1. The Maestro Main Application Window

The functions of the Main Application Window components are as follows:

- **Workspace**—displays drawn or read-in structures
- **Main Menubar**—provides panel access via pull-down menus
- **Sidebar**—contains structure manipulation and pick state selection tools
- **Auto-Help**—displays automatic, context-sensitive help
- **Top View**—displays Workspace structures as viewed from above
- **Command Input Area**—provides a way to execute command line operations from within the Maestro interface
- **StatusBar**—displays the number of atoms, entries**,** residues, chains and molecules currently displayed in the workspace

3.2 The Project Facility

Central to Maestro is the concept of a *project*. There is always a project open in Maestro. If you do not specify a project when you start Maestro, Maestro creates a *scratch* project. You can work in a scratch project, but you must save it in order to use it in future sessions.

A project is a collection of chemical structures (“Entries”) and their associated data. These structures and their data are graphically represented in the Project Table, which displays an ordered list of entries and any associated data. You can open the Project Table panel by choosing Show Table from the Project menu.

Entries are represented by rows in the Project Table. Each row contains the row number, the name assigned to the structures represented by the entry, the entry’s Workspace inclusion state, any title associated with the entry, and a button to access the Surfaces panel if there are surfaces associated with the entry. If there are no surfaces associated with an entry, the Vis column of the Project Table remains empty.

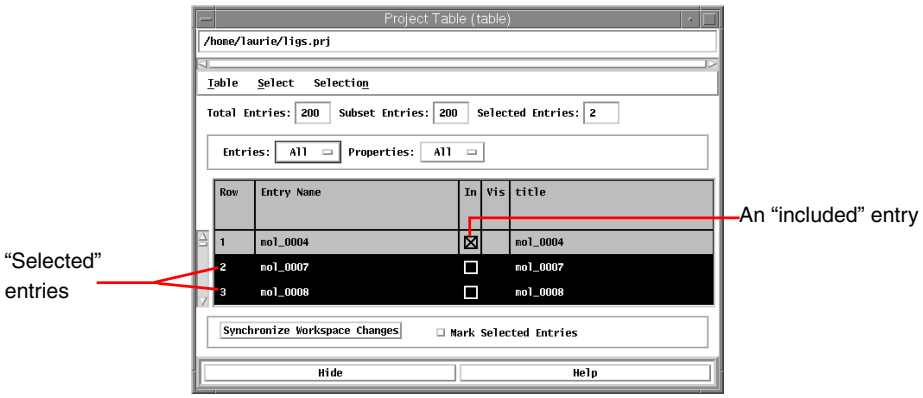

Figure 3.2. The Project Table ***panel***

The Project Facility allows you to operate on these entries. You can choose which entries to include in the Workspace by checking the corresponding In box on the Project Table panel. Including an entry causes any structures associated with that entry to be rendered in the Workspace. You can exclude a structure by clearing the check box.

You can select an entry or entries to use as input for a QikProp job, or you can use the entry or entries in the Workspace. In addition, you can select entries as input for the ePlayer, which displays, in sequence, the selected structures. Additionally, you can select entries on which to operate using other features such as “Duplicate,” “Combine,” “Rename,” or “Sort.”

When you build molecules in the Workspace, they constitute the *scratch* entry until you choose to save the structures as project entries. The scratch entry is not saved with the project unless you explicitly incorporate it into the project. However, you can use a scratch entry as input for some computations.

3.3 Maestro Workflow Paradigm

The Maestro interface uses a "define operation, then pick atoms or bonds" workflow paradigm. That is, the action to be performed must be selected and set up before the atoms to be operated upon can be designated. For example, to change the color of all atoms in an on-screen structure to the color "green," you first select the color from the Atom Coloring panel, and then pick the atoms to be colored.

3.4 Picking Atoms

Once an operation has been specified and set up, atoms must be selected. The name for this process is "picking." Picking is accomplished by clicking on desired structural elements in a Workspace structure. The way in which Maestro interprets each mouse click is dependent on the *pick state* and the *picking mode*.

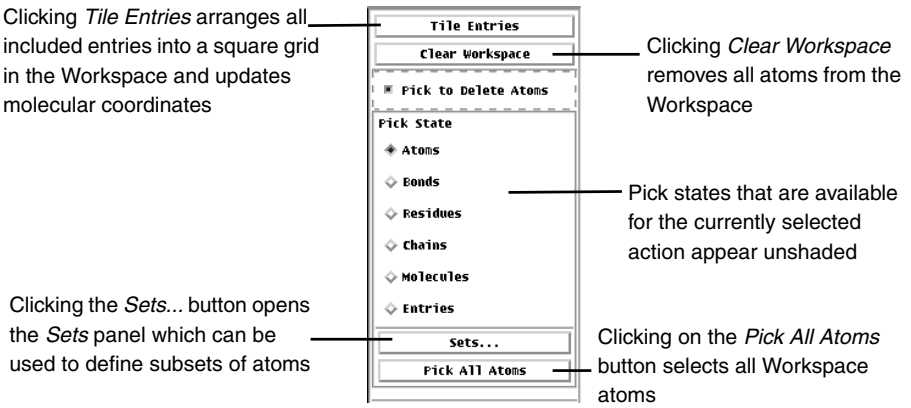
Clicking *Tile Entries* arranges all included entries into a square grid in the Workspace and updates molecular coordinates
Tile Entries
Clear Workspace
Clicking *Clear Workspace* removes all atoms from the Workspace
Pick to Delete Atoms
Pick State
Atoms
Bonds
Residues
Chains
Molecules
Entries
Pick states that are available for the currently selected action appear unshaded
Clicking the *Sets...* button opens the *Sets* panel which can be used to define subsets of atoms
Sets...
Pick All Atoms
Clicking on the *Pick All Atoms* button selects all Workspace atoms

Figure 3.3. The Pick State ***menu on the Main Application Window Sidebar***

3.4.1 Pick States

Maestro pick states allow you to specify how a "pick" is interpreted. A click on a single atom can represent either the selection of only that atom or the selection of the residue, chain, molecule, or entry of which it is a member. You change the pick state by clicking the radio button that corresponds to your choice in the Pick States area of the Sidebar.

3.4.2 Picking Mode

Maestro panels are amodal—more than one panel can be open at any given time. A panel need not be closed for an action to be carried out. You might find it convenient to leave frequently used panels open.

In contrast to modal panels, which carry out designated actions only upon closing, actions occur immediately in amodal panels once an operation and atoms have been specified. To make you aware of this difference in functionality, there is no Close option on the panels' pull-down menu. Instead, there is a Hide button in the lower left corner of each panel.

Since more than one panel can be open at a time, the picking focus—the designation of the panel that is intended to "receive" the picked atoms—must be moved from one panel to another. When a new panel is opened, it automatically receives picking focus. To return to an already open panel and give it picking focus, you must move the keyboard focus to that panel and then enable the Pick to <action> toggle button, where <action> is the operation for which you would like to pick atoms. Moving the picking focus is called changing the "picking mode."

3.5 Mouse Functions

Within Maestro, each button of a three-button mouse has a fixed function. A single click with the left button is always used for selecting. This includes picking atoms, bonds, and atom groups; clicking buttons; and selecting menu items. The middle mouse button is used to rotate molecules. The right button is always used for translating atoms or molecules along the x, y, or z axis. Right-clicking on an atom in the Workspace causes that atom to be spot-centered in the Workspace. Holding down the middle and right mouse buttons in combination is used to zoom in and out on a molecule.

If you have the handedness on your mouse set to "left," your mouse functions are the mirror image of those described: your right mouse button is used for picking, and your left button is used for translating.

3.5.1 Specialized Mouse Functions in the Project Facility

Clicking on project entries with the mouse while pressing the Shift key or the Control key allows you to select any combination of entries. For instance, clicking on an entry while pressing the Control key toggles that entry's selection status. If an entry is already selected, clicking on an additional entry while pressing Shift will cause all entries listed between the two to be selected as well. However, there are "shortcuts" for selecting some entry groups on the Select pull-down menu and the Select panel. Open the Select panel from the Select pull-down menu on the Project Table panel.

Once you have selected project entries, you can manipulate them using tools accessed from the Selection pull-down menu. Clicking the right mouse button while the pointer is positioned over the Project Table (so long as there is at least one entry in the table) displays the Selection menu at the mouse position.

You can use the mouse to perform operations besides selecting entries. For instance, clicking on the corner of the Project Table panel and dragging resizes the panel. Clicking and dragging with the middle mouse button on the boundary of a row resizes that row. Finally, clicking on selected entries and dragging allows you to reposition the entry in the Project Table. The entries are placed after the first unselected entry that precedes the entry on which the cursor is resting when you release the mouse button. For example, if you select entries 2, 4, and 6, and release the mouse button on entry 3, these three entries are placed after entry 1, because entry 1 is the first unselected entry that precedes entry 3. To move entries to the top of the table, drag them above the top of the table; to move entries to the end of the table, drag them below the end of the table.

A summary of project-based mouse functions is provided in the following table:

Task	**Mouse Operation in the Project Table**
Entry selection	Left-click
Display of Selection menu	Right-click
Multiple entry selection	Shift-click
Toggles entry selection status	Control-click
Move entry	Click+drag

3.6 Sidebar Controls

The Main Application Window Sidebar contains controls related to picking, to saving and restoring the viewing transform, and to molecular coordinate transformations (translations and rotations). To specify an axis other than the default value of *xy* for translations or rotations, select the option corresponding to the desired axis from the Rotate or Translate option menu. To specify whether you would like all atoms in the Workspace to be rotated or translated, or just a subset, enable the appropriate toggle button (Global to move all atoms or Local to move only selected atoms). These buttons appear on the Advanced Transformation panel, which you open using the Advanced... button on the Sidebar.

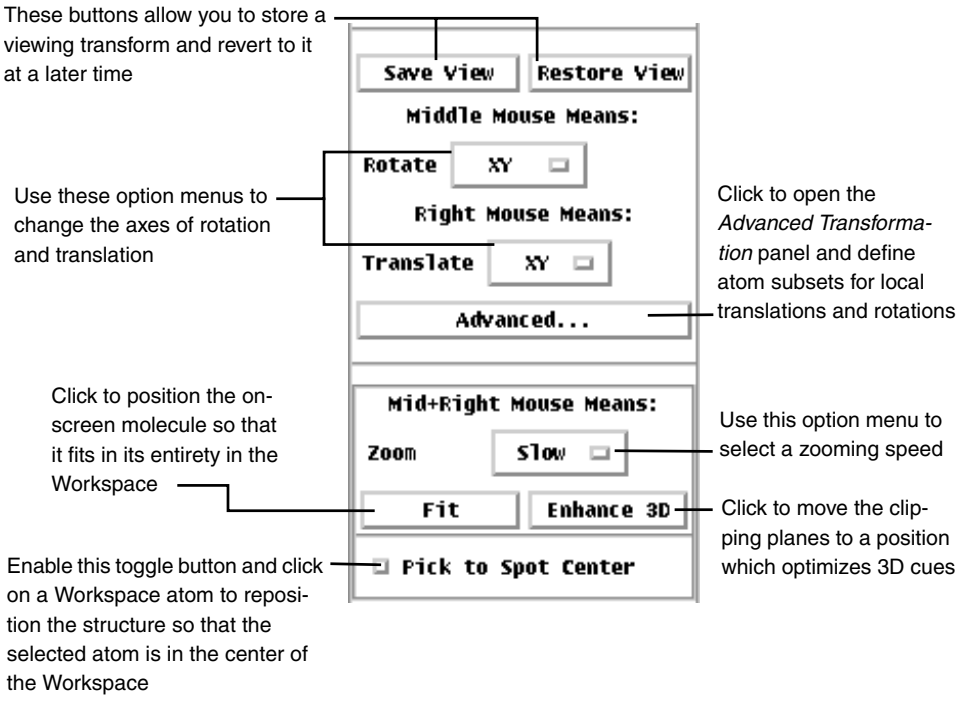
These buttons allow you to store a viewing transform and revert to it at a later time
Save View
Restore View
Middle Mouse Means:
Rotate
XY
Use these option menus to change the axes of rotation and translation
Right Mouse Means:
Translate
XY
Advanced...
Click to open the *Advanced Transformation* panel and define atom subsets for local translations and rotations
Mid+Right Mouse Means:
Zoom
Slow
Use this option menu to select a zooming speed
Click to position the on-screen molecule so that it fits in its entirety in the Workspace
Fit
Enhance 3D
Click to move the clipping planes to a position which optimizes 3D cues
Pick to Spot Center
Enable this toggle button and click on a Workspace atom to reposition the structure so that the selected atom is in the center of the Workspace

Figure 3.4. Transformation controls on the Main Application Window Sidebar

3.7 Maestro Command Scripts

Although there is a provision for performing nearly all Maestro-supported operations by interacting with menus and panels, the operations can also be performed using Maestro commands, or compilations of these commands, called scripts.

Command scripts can be used to automate lengthy procedures or repetitive tasks. And, because all Maestro commands are logged and displayed on the Command Script Editor panel, it is not necessary to memorize the commands for every operation. Simply perform the desired operations once using the menus and panels within Maestro, copy the logged commands from the Command History list into the Script area of the panel, and then save the list of copied commands.

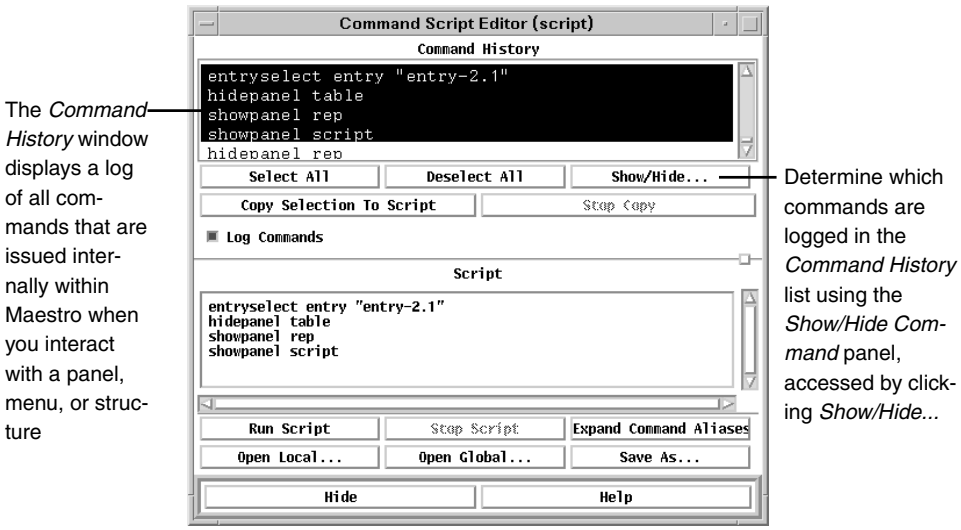

Figure 3.5. The Command Script Editor ***panel***

To run an existing command script, first open the Command Script Editor panel from the Edit pull-down menu in the Maestro Main Application Window. Then, click Open Local... and navigate to the directory containing the desired script. Click the name of a script in the Files list, and then click Open. The command script is loaded into the Script window of the Command Script Editor panel. To execute the script, click Run Script.

Chapter 4: Launching the Programs

4.1 Introduction

QikProp, QikFit, and QikSim can all be run from the command line. QikProp can also be run from the Maestro GUI. Using Maestro to run QikProp gives you the advantage of being able to build structures or import structures into Maestro, use them as input for a QikProp job, and then use Maestro's analysis tools and Schrödinger's other computational programs to further investigate lead compounds. Whether you run QikProp from the Maestro GUI or from the command line, you must set the SCHRODINGER environment variable.

This chapter describes how to launch Maestro and open the QikProp panel, and how to submit QikProp, QikFit, and QikSim calculations from the command line.

4.2 Environment Variables

Before you can run Maestro, QikProp, QikFit, or QikSim, you must first set the SCHRODINGER environment variable. In a terminal window, use the appropriate version of the following command to set the SCHRODINGER environment variable to point to the directory in which Maestro and QikProp are installed:

csh: `setenv SCHRODINGER <installation-directory-path>`

bash or ksh: `export SCHRODINGER=<installation-directory-path>`

To avoid setting the SCHRODINGER environment variable manually each time you launch Maestro, QikProp, QikFit, and QikSim from a new terminal window, you can add the appropriate version of the above command to your login shell script file (*.cshrc* or *.login* for csh, *.profile* for ksh, or *.profile* or *.bashrc* for bash).

Also for convenience, you might want to add $SCHRODINGER to your $PATH definition. This allows you to run Maestro, QikProp, QikFit, and QikSim by simply typing their script names in a terminal window once SCHRODINGER has been set.

In some cases, the DISPLAY environment variable is set automatically upon login. If it isn't, you must either set it automatically by adding the DISPLAY variable definition to your login shell script (*.cshrc* or *.login* file for csh or *.profile* file for bash or ksh), or set it manually each time you launch the program from a new terminal window. To set the DISPLAY environment variable use the appropriate version of the following command, substituting the name of the display machine for <machine_name>. (To determine the name of a display machine, type `hostname` in a terminal window.)

csh: `setenv DISPLAY <machine_name>:0.0`

bash & ksh: `export DISPLAY=<machine_name>:0.0`

4.3 Launching QikProp from Maestro

Before you can run QikProp from Maestro, you must first launch a Maestro session. To launch QikProp from Maestro:

1. In a terminal window, enter the command:

   ```
   $SCHRODINGER/maestro
   ```

2. In Maestro's Main Application Window, choose QikProp... from the Applications menu. The QikProp panel is displayed. For information about submitting a QikProp calculation using this panel, see Chapter 6: "Maestro QikProp" on page 41.

4.4 Launching Command Line QikProp

In a terminal window, enter the command:

```
$SCHRODINGER/qikprop [<options>] <input_file>
```

replacing [<options>] with any desired options and replacing <input_file> with the name of the structure file to be evaluated.

If the file resides in a directory other than the current working directory, specify the path as well as the file name. For more information about command line QikProp and the valid options, see Chapter 8: "Command Line QikProp" on page 51.

4.5 Launching QikFit

In a terminal window, enter the command:

```
$SCHRODINGER/qikfit [<options>] <training_file>
```

replacing [<options>] with any desired options, and replacing <training_file> with the name of the comma-separated value (CSV) file to which you have added a new column of experimental activity data. For more information about QikFit, see Chapter 9: "QikFit" on page 55.

4.6 Launching QikSim

In a terminal window, enter the command:

```
$SCHRODINGER/qiksim [<options>] <CSV_file>
```

replacing [<options>] with any desired options, and replacing <CSV_file> with the name of the CSV file that you want to evaluate. For more information on QikSim, see Chapter 10: "QikSim" on page 61.

4.7 Displaying Command Syntax

The start-up script of each of these QikProp programs prints a brief summary of the available options and appropriate command syntax if the script is executed with no input file specified. For example, typing:

```
$SCHRODINGER/qikprop
```

prints out a synopsis of QikProp options.

Chapter 5: QuickTopics

5.1 Introduction

The "QuickTopics" chapter of this document is designed to assist you in rapidly becoming familiar with the functionality of QikProp 2.0. Once you have worked through these exercises, you will have an understanding of the basic QikProp, QikFit, and QikSim features.

This chapter assumes that you have already installed the Schrödinger software, or that you have access to an installation of the Schrödinger software. If you do not have access to the software, you must install it. For installation instructions, see the *Schrödinger Product Installation Guide*. It is also assumed that you have set the SCHRODINGER environment variable according to the instructions in section 4.2 "Environment Variables" on page 15.

5.2 Getting Ready to Use QikProp from Maestro

Before you can run QikProp from Maestro, you must first launch a Maestro session. To do so, use the instructions below. QikFit and QikSim have not yet been interfaced with Maestro, so it is not necessary to launch Maestro to use these modules of the QikProp program.

Launching Maestro

Use the instructions in Chapter 4: "Launching the Programs" on page 15 to set the SCHRODINGER environment variable to point to the directory in which Maestro and QikProp are installed, and then to launch Maestro.

Specifying a Maestro Working Directory

When you use Maestro to launch QikProp jobs, Maestro writes job output to the directory specified in the Directory folder of the Preferences panel. By default, the file i/o directory (the directory to which Maestro will write files) is the directory from which you launched the Maestro program. If you want output files to be placed in another directory, you can change the preferences as described below.

1. Open the Preferences panel from the Maestro menu in the Maestro Main Application Window.
2. Click the Directory folder.
3. Click the radio button corresponding to the directory you want files to be read from and written to.

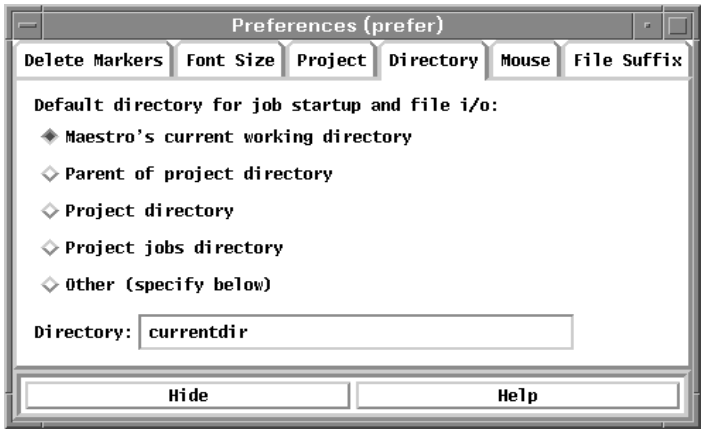

Figure 5.1. The Directory ***folder of the*** Preferences ***panel***

Creating a Working Tutorial Directory

In the **$SCHRODINGER/qikprop-v<version>/tutorial** directory of your QikProp distribution are two Maestro-formatted multi-structure files that you will use in the following exercises. Before you begin the tutorial, you must create a directory in which to write the files that are produced during the exercises. To create the directory, open a terminal window and do the following:

1. Use the `cd` command to move to a directory in which you have write permission.
2. Create a directory called **tutorial** by typing the command:

   ```
   mkdir tutorial
   ```

   In the text that follows, we refer to this directory as the "working directory."

3. Set the SCHRODINGER environment variable to the installation directory.
4. Copy the files *ligs.mae*, *test.mae*, and *data.txt* to your working directory by entering the following command at the prompt. Replace <version> with the version number of your particular QikProp release.

   ```
   cp $SCHRODINGER/qikprop-v<version>/tutorial/*.* .
   ```

5.3 QuickTopic Exercises

Importing Structures to Use as QikProp Input

When running QikProp from Maestro, you can use as input for a job either the structure(s) rendered in the Workspace or entries that are selected on the Project Table panel. In this exercise, you will import structures into Maestro to use as QikProp input.

1. Choose Import Structures... from the Project menu in the Main Application Window. The Import panel is displayed.

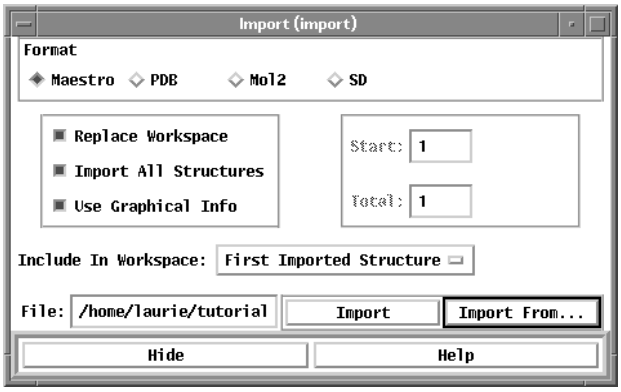

Figure 5.2. The* Import *panel

2. Click Import From... to bring up the Import Structure Files file selector.
3. Select <*working_directory*>*/ligs.mae.*
4. Click Open.
5. Choose Show Table from the Project menu in the Main Application Window. The Project Table panel is displayed. The contents of the *ligs.mae* file are listed, and each of the 200 entries are selected.

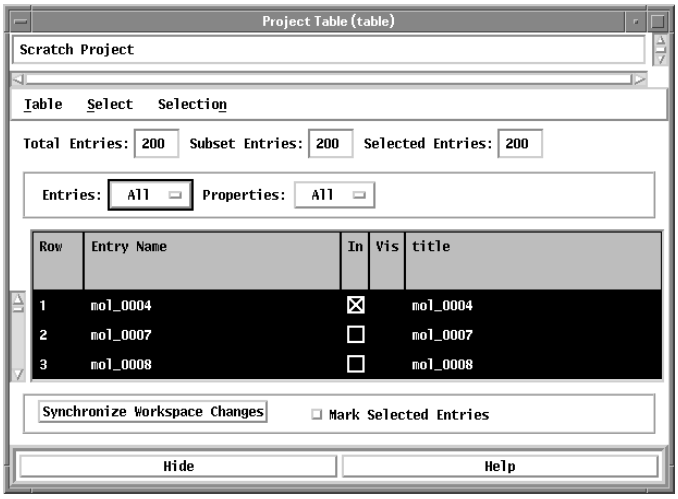

Figure 5.3. The Project Table ***panel with all entries selected***

In the next few exercises you will use QikProp to make predictions for this set of structures.

Using QikProp to Generate a Training Set

In this exercise, you will run QikProp in normal processing mode to create an input training file for a QikFit job which will be run in a later exercise.

1. In the Main Application Window, open the QikProp panel from the Applications menu.
2. In the Job: text box of the QikProp panel, type `ligands`.

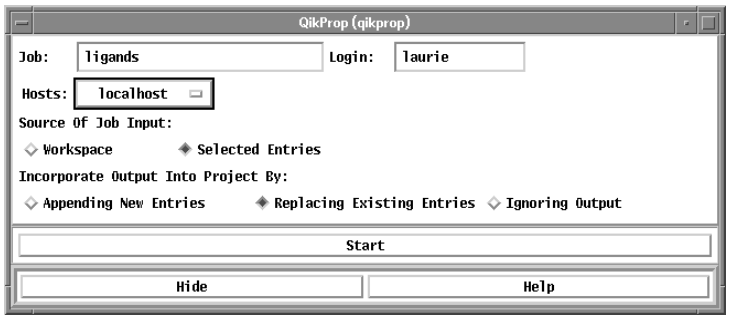

Figure 5.4. The** QikProp **panel

3. Choose Selected Entries for the Source of Job Input: setting.
4. Choose Replacing Existing Entries under the Incorporate Ouput Into Project By: heading.
5. Click Start. The Hydrogen/Lone-Pair Fixup panel is displayed.

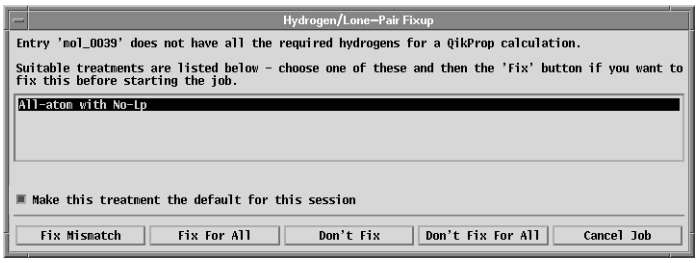
Hydrogen/Lone-Pair Fixup
Entry 'mol_0039' does not have all the required hydrogens for a QikProp calculation.
Suitable treatments are listed below - choose one of these and then the 'Fix' button if you want to fix this before starting the job.
All-atom with No-Lp
Make this treatment the default for this session
Fix Mismatch
Fix For All
Don't Fix
Don't Fix For All
Cancel Job

Figure 5.5. The** Hydrogen/Lone-Pair Fixup **panel

You must choose to fix the input, because QikProp results are not valid for input structures that are missing hydrogens. Click Fix For All. The Monitor panel is displayed and the job begins running.

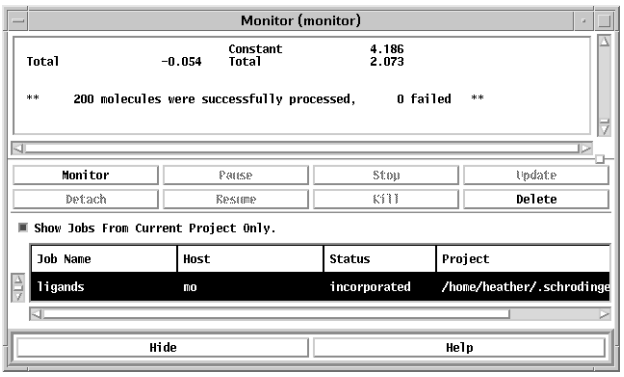

Figure 5.6. The Monitor ***panel displaying output from the ligands job***

When the job is complete, the following files should be in your working directory:

- *ligands-out.mae*
- *ligands.mae*
- *ligands.out*
- *ligands.qpsa*
- *ligands.CSV*
- *ligands.warning*

The file *ligands-out.mae* is a Maestro-formatted file that contains property data and structural information. The *ligands.mae* file is written by QikProp at the start of the job and contains only structural information for the selected entries. The *ligands.out* file lists properties predicted and the limits specified in *QPlimits*. It also includes a section that shows how some of the properties were formed by linear combinations of more basic molecular properties. The *ligands.qpsa* file provides a detailed surface area analysis, and *ligands.CSV* contains all the predicted properties for each structure in comma-separated value (CSV) format.

The <*filename*>.*warning* file is produced only when QikProp encounters problems with one or more of the input structures. In this test case, all structures were processed successfully. However, results for three of these structures might not be valid because one of the processed structures is not neutral and two of the structures contain ammonium ions. For more information on output files see Chapter 7: “QikProp Output” on page 47.

6. Save a copy of the project by choosing Copy Project To... from the Project menu in the Main Application Window. Name the project *ligs.prj* and click Copy.

Running QikProp in Fast Processing Mode

In this exercise, you will run QikProp in fast mode. You will use the same set of structures you processed in normal mode in the previous exercise so that you can compare the output files produced by each mode.

1. Choose Save Scratch Project... from the Project menu of the Main Application Window, enter `comp.prj` under Project, and click Rename. The current scratch project is now named ***comp.prj***.
2. Copy the file *QPlimits* from the **$SCHRODINGER/qikprop-v<version>/data** directory into the working directory.
3. Open the *QPlimits* file with a text editor. Below is an excerpt from the file.

```
Molecular Weight                      standard is:  ( 130.0 / 725.0)
 130.0 / 725.0
Dipole Moment (D)                     standard is:  (   1.0 /  12.5)
   1.0 /  12.5
...
QP log Khsa Serum Protein Binding     standard is:  (  -1.5 /   1.2)
  -1.5 /   1.2
QP log BB for     brain/blood         standard is:  (  -3.0 /   1.0)
  -3.0 /   1.0
Mode (std or fast)
std
```

   The first portion of the file contains the default limit settings. These can be edited to suit your needs. In this exercise you do not need to modify the defaults.

4. Scroll down to the end of the file. Locate the line that begins with `Mode`. Replace the text on the next line with the word `fast`. Save the change and close the file.
5. Ensure that all entries in the Project Table panel are selected, then open the QikProp panel from the Applications menu in the Main Application Window.
6. In the QikProp panel's Job: text box, type `QPfast`. For Source of Job Input: choose Selected Entries and under Incorporate Output Into Project By: choose Appending New Entries.

7. Click Start. The job runs in a few seconds. When it finishes, the Project Table contains 400 entries.

Sorting Project Table Entries

In this exercise, you will use the Project Table panel's sorting function to help compare the results from fast mode to the results from normal mode.

1. In the Project Table panel, choose Select All from the Select menu.
2. Choose Sort... from the Table menu. The Sort Project Table panel is displayed. Ensure that Ascending is selected for the Order: option.
3. Choose Entry Name and click Sort selected table rows. The row for each fast mode entry is now directly underneath the row for the corresponding normal mode entry.

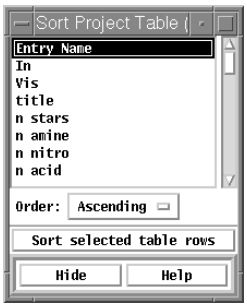

Figure 5.7. The Sort Project Table ***Panel***

4. Scroll through the Project Table and compare the fast mode results with the normal mode results. Notice that, in fast mode, the dipole moment ("dipole"), dipole^2/volume ("d2ov"), the ionization potential ("ip"), and the electron affinity ("ea") are not predicted. Instead 0.0 is entered for these values. This is because, in the interest of saving time, the PM3 single point calculation is not performed. Notice also that the predictions for the free energy of solvation terms ("log pc16," "log pw," and "log po w") differ between fast and normal processing modes because, in fast mode, the QSPR equations are refit without the dipole terms.

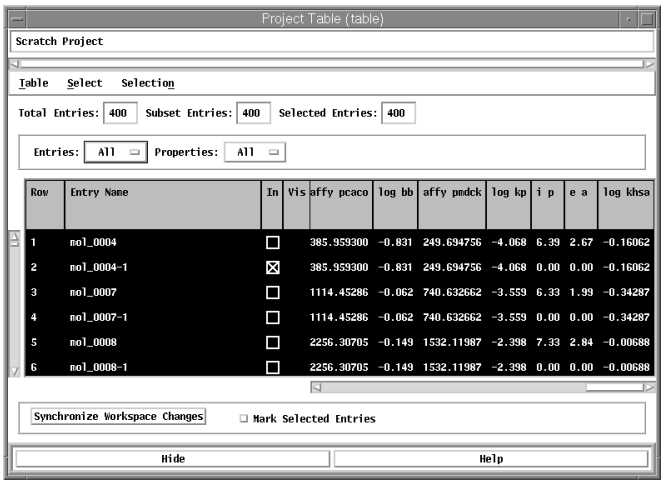

Figure 5.8. The Project Table with fast and normal mode properties sorted

5. In subsequent exercises, you will run QikProp in normal mode so you must edit the *QPlimits* file again. Open the *QPlimits* file with a text editor and replace the word `fast` after the `Mode` line with the original text. (It does not matter what text replaces `fast`.) Save the change and close the file.

6. Close the current project by selecting Close Project from Maestro's Project menu.

Creating a Training File for use with QikFit

To investigate how to use the QikFit program, you must first generate a training file. During this exercise, you will use the Microsoft® Excel spreadsheet program to create the file. Ordinarily, when creating training files for use with QikFit, you would add a column of new experimental data. For use with this exercise, a list of data has been provided in the **tutorial** directory. Using the instructions in this exercise, copy the list into the CSV file. In the next exercise, you will then use QikFit to fit to this data.

These instructions are specific to Excel, but another, comparable spreadsheet program can be used.

1. Launch Excel, and from the File menu, select Open.
2. From the Files of Type option menu, select Text Files.
3. Navigate to the directory containing your *ligands.CSV* file, and double-click on the file name. Note: If file extensions are not visible on your computer, *ligands.CSV* is visible simply as *ligands*.
4. Open the file *data.txt*. The Text Import Wizard is displayed. Click Finish. It is not necessary to specify a delimiter because the file contains only a single column of data. A column of data named "exp_data" is displayed. This is our experimental property.
5. Copy the column of data, and paste it into the first empty column in *ligands.CSV* (just to the right of the column that contains data for "QPlogKhsa").
6. Save the file as *training.CSV* by selecting Save As from the File menu and entering `training.CSV` in the text box. Ensure that CSV is selected from the Save as type option menu.
7. Click Save and close Excel.

Using QikProp to Generate a Test File

QikFit can calculate the experimental property for molecules in a test file whose name is specified on the command line. At the same time, it can produce a file containing regression coefficients for the experimental value. In this exercise, you will use QikProp to generate the <*jobname*>*.CSV* file for a molecule called mol_7147, for which there is no "exp_data" value. Instead of running QikProp through the Maestro interface, you will submit the job from the command line. Generally, the command line option is intended for running jobs that consist of large numbers of structures. To create a test file for use with QikFit, enter the following command in a terminal window:

```
$SCHRODINGER/qikprop test.mae
```

When the job is complete, the following files should be in your working directory:

- *test-out.mae*
- *test.mae*
- *test.out*
- *test.qpsa*
- *test.CSV*

You will use the *test.CSV* file in the QikFit run in the following exercise.

Running QikFit

In this exercise, you will use QikFit to predict the new "exp_data" property for the test structure mol_7147, whose other properties were predicted in the previous exercise.

1. To predict the "exp_data" property, run QikFit by typing the following command in a terminal window:

```
$SCHRODINGER/qikfit -t test.CSV training.CSV
```

By default, QikFit treats the last column in the training file as the new experimental value. If you want to specify a different column, you can do so with the `-a` option.

When the job is finished, the following files should be in the directory from which you launched QikFit:

- *training.myfits*
- *training.out*
- *training.log*

The file *training.out* includes the parameter values calculated during each cycle of the regression, statistical data for each cycle, and a list of the experimental and calculated values for each compound. An excerpt from the *training.out* file is shown below.

```
...
      N    Exptl Activity  Calculated    Delta      Compound
      1     1.331762      1.206324      0.125437    mol_0004
      2     1.217879      1.142674      0.075205    mol_0007
      3     0.248979      1.046550     -0.797572    mol_0008
...
    199     0.856038      1.145599     -0.289561    mol_3085
    200     1.838763      1.177887      0.660876    mol_3138
...
 Final Parameter Values

 Descriptor           Parameter    Significance   Partial F    Partial R^2
  #stars               0.000000       0.0000        0.0000       0.0000
  #amine               0.000000       0.0000        0.0000       0.0000
...
```

```
  QPpolrz              0.052912       0.0290        6.1028       0.0299
  QPlogPC16           -0.182341       0.0267        2.5284       0.0126
...
               New Predictions

      N    Calc Activity    Compound
      1     1.050474        "mol_7147"
```

The first section shown displays the molecule number, experimental property value, and property value calculated by the regression equations for each structure. The `Delta` column shows the difference between the experimental and calculated values, and the `Compound` column gives the molecule name for each structure.

The next section shown gives, along with the significance, the parameter or coefficient calculated for each property. The partial F and partial R^2 are reported for each property. Properties that were eliminated from the regression are assigned values of `0.000000` for all fields.

The last section, "New Predictions," reports the calculated value for the test structure, mol_7147.

For a more detailed explanation of the QikFit *.out* file, refer to section C.1 "Sample Regression Output and Commentary" on page 81.

Predicting New Properties with the QPmyfits File

While QikFit can predict new properties for a molecule in a test file during a run, it is likely that you will want to use the subsequent <*jobname*>*.myfits* file to allow QikProp to begin predicting the new property along with the 36 default properties and descriptors for several molecules at once.

The *training.myfits* file that was generated in the last exercise contains the coefficients necessary for QikProp to predict "exp_data" for molecules that are structurally similar to those molecules included in the file. However, in this exercise, for cross-reference purposes, we will use *training.myfits* with the same set of training molecules—those from the file *ligs.mae*—that were evaluated in the previous exercises. Follow the instructions below to use *training.myfits* in a QikProp run.

1. Rename the *training.myfits* file *QPmyfits* and ensure that the *QPmyfits* file is in the current working directory.
2. Import *ligs.mae* into Maestro.
3. Open the QikProp panel from the Applications menu. Type `ligands2` in the Job: text box and under Incorporate Output Into Project By:, again choose Replacing Existing Entries.
4. Click Start. The Hydrogen/Lone-Pair Fixup panel is displayed. Choose Fix For All.

This job produces the files *ligands2-out.mae*, *ligands2.out*, *ligands2.CSV*, and *ligands2.qpsa*. This time an additional property appears in the Project Table under the heading "myfit1." Look at this data and compare the predicted "exp_data" values with the originals in *training.CSV*. Note that the predictions that appear in *ligands2.CSV* are the same as the calculated values reported for each structure in *training.out*.

Creating a Plot of Project Table Properties

In the next few exercises, you will use the plotting capabilities of the Project Table to examine data produced by QikProp in the previous exercise. In this particular exercise, you will plot data for a subset of Project Table entries. In the next exercise, you will create a different type of plot.

1. In the Project Table panel, select a subset of entries to plot by clicking on entry 1, holding down the Shift key, and then clicking on entry 20. Entries 1 through 20 should now be selected.
2. Choose New Plot... from the Selection menu. The New Series dialog box is displayed.
3. To visualize the range of solvent accessible surface areas (SASAs) in the selected set of entries, you will plot "SASA" versus "Entry Name." In the New Series dialog box, under Plot Name: type SASA. For the x axis property, leave Entry Name selected in the X Property box. For the y axis property, choose sasa.

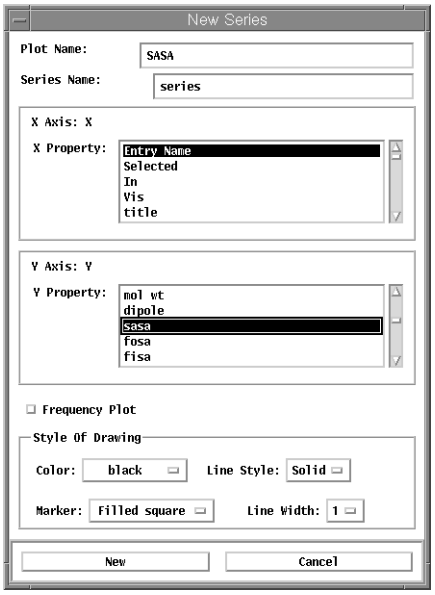

Figure 5.9. The* New Series *dialog box

4. Click New. The plot is displayed in the Plot XY panel, as shown

below. Keep this panel open for use in the following exercise in which you will add a scatter plot.

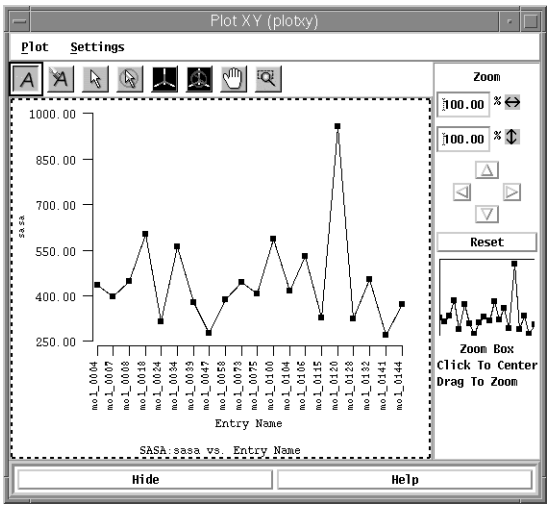

Figure 5.10. The Plot XY ***panel***

Creating A Scatter Plot

In this exercise, you will create a scatter plot of data for all entries in the Project Table, and you will explore some additional features of the Plotting Facility.

1. In the Project Table panel, choose Select All from the Select menu.
2. In the PlotXY panel, choose New Plot... from the Plot menu. The New Series dialog box is displayed. This time you will create a scatter plot to visualize the correlation between two properties. Under Plot Name: enter `Scatter`. For the x axis property, select mol wt, and for the y axis property, select volume.
3. Change the plot style by making selections in the Style of Drawing section. For Color: choose blue, under Line Style: select None, and for Marker select Cross. Click New to generate the new plot.

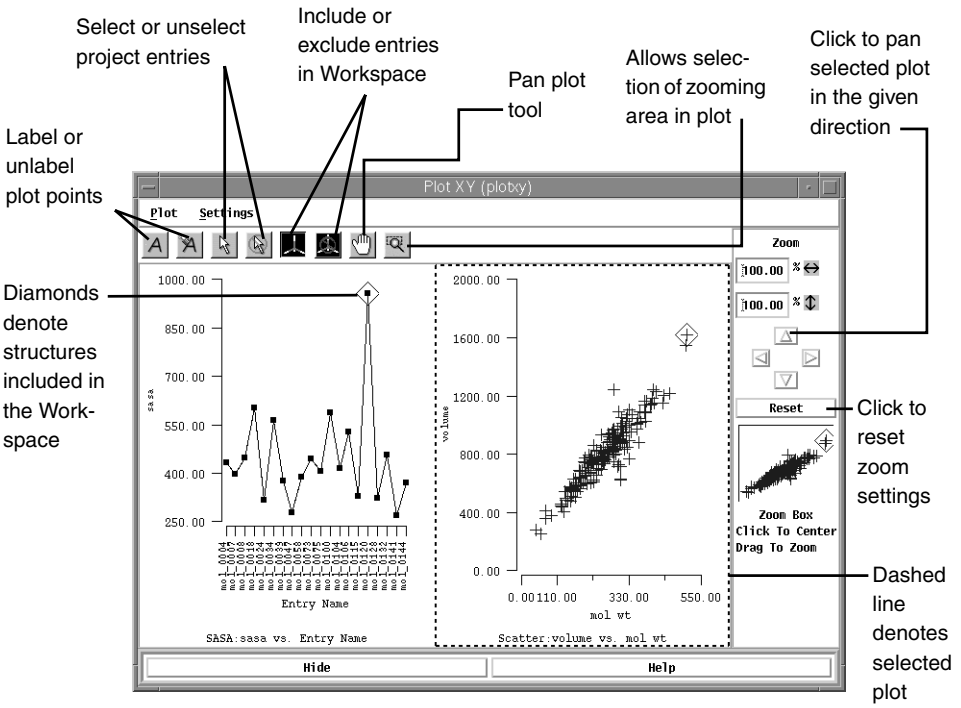

Figure 5.11. Plot XY ***panel***

The icons at the top of the PlotXY panel can be selected to change the mouse function within that panel. You can choose to select or include Project Table entries in the Workspace by clicking on points within a plot. For instance, clicking on the icon in the toolbar of the Plot XY panel, and then clicking on the point in the plot which has the largest SASA value causes a diamond-shaped marker to appear around the point in the plot. The structure is also included in the Workspace. Additional structures can be included in the Workspace by holding down the SHIFT key and clicking on other points in the plot. These can be excluded from the Workspace by choosing the icon in the toolbar and clicking points in the plot.

The dashed line surrounding a plot indicates that it is selected. You can select plots by clicking the arrow icon in the tool bar, and then clicking on the plot. Once a plot is selected, you can perform various operations on it. To view only the new plot, which is the only one currently selected, choose Display Only Selected Plots from the Plot menu. To view both plots again, use the Plot menu to choose first Select All and then Display Selected Plots. Both plots will then be part of the project and will be saved with the structural and property data.

In the subsequent exercise, you will execute QikSim from the command line, so you may now close Maestro.

Running QikSim and Examining Output Data

QikSim ranks molecules based on structural similarities to a specified reference molecule. As the basis for comparison, QikSim uses the Euclidean distance and the Tanimoto coefficient. To perform a QikSim calculation, you must specify a reference or "probe" structure. The first structure will be used by default, but any structure in the file can be specified with the `-p` option. Use the following instructions to perform the QikSim calculation.

1. Run a QikSim calculation by typing the following command in a terminal window:

   ```
   $SCHRODINGER/qiksim -p 10 ligands.CSV
   ```

   The `-p` option selects the tenth structure as the probe molecule. All other structures are compared to structure ten. The job should run in a few seconds.

2. View the contents of *ligands.simout*. An excerpt from this file is shown below:

```
...
     Euclidean Distance                  Tanimoto Coefficient
   N       R^2         Name               N        TC        Name
   10      0.000 mol_0073                 10      1.000 mol_0073
  130      0.091 mol_1191                130      0.997 mol_1191
   24      0.198 mol_0159                 18      0.985 mol_0132
   18      0.514 mol_0132                 24      0.983 mol_0159
   11      0.744 mol_0075                 11      0.963 mol_0075
...
```

The output file should show mol_0073, the tenth structure in the input file, as the first structure for which the Euclidean distance and the Tanimoto coefficient are reported. The Euclidean distance for this molecule should be 0.000 and the Tan-

imoto coefficient should be 1.000 because the molecule is being compared with itself. The next closest molecule is mol_1191 according to both the Euclidean distance and the Tanimoto coefficient.

5.4 Summary

You have completed the QuickTopic exercises. More information about each program—Maestro, QikProp, QikFit, and QikSim—can be found in the corresponding chapters of this manual.

Chapter 6: Maestro QikProp

6.1 Opening the QikProp Panel

To open the QikProp panel from Maestro, first launch Maestro according to the instructions in Chapter 4: "Launching the Programs" on page 15. Then select QikProp... from the Applications menu. The QikProp panel is displayed.

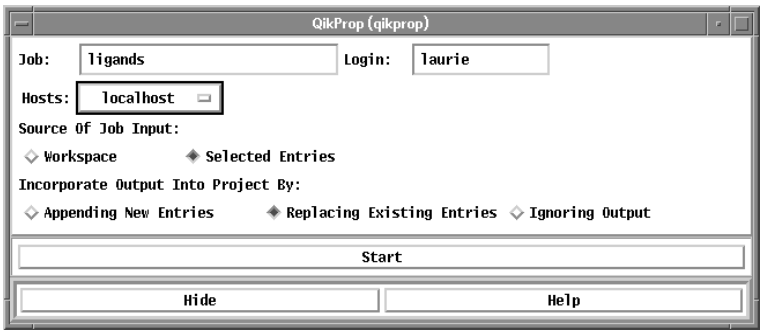

Figure 6.1. The* QikProp *panel

To launch a job from the QikProp panel, you must first specify a job name, a host, an input source, and an incorporating mechanism. If you wish to use fast processing mode, you must also edit the *QPlimits* file.

6.2 Naming the QikProp Job

By default, when the QikProp panel is first opened, the job name `qikproptmp` appears in the Job: text box. You can accept this default name and run a calculation, but Maestro's Job Control Facility does not allow you to run multiple jobs with the same name. To run a job that has the same name as a previous job, you must first remove files associated with the first job.

6.3 Selecting a Host Computer

If you want to run QikProp locally, you do not need to configure your system. If you want to run a QikProp job on a remote computer, you must do the following four steps.

1. Prepare the computers and accounts
2. Create a *schrodinger.hosts* file
3. Restart Maestro
4. Select the desired host from the Hosts: option menu on the QikProp panel

The first three steps need to be performed only once after Maestro is installed, unless more hosts are added. The fourth step must be performed each time a job is launched on a remote host. The first two steps are described in section A.3 “Preparing for Remote Job Submission” on page 70 and section A.4 “The schrodinger.hosts File” on page 71.

When you submit a QikProp job from Maestro, use the Hosts: option menu on the QikProp panel to select a host on which to run the job. All hosts that have been configured in the *schrodinger.hosts* file appear on this menu.

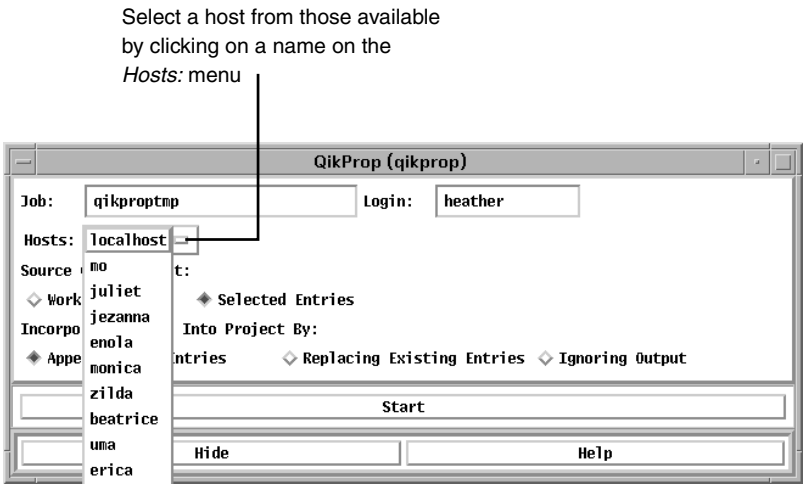

Figure 6.2. The* Hosts: *menu on the* QikProp *panel

6.4 Using the QPlimits File

The *QPlimits* file can be used to run QikProp in fast processing mode or to customize the ranges of values that determine which molecules are to be flagged as being dissimilar to other known drugs. The file can be copied from the **$SCHRODINGER/qikprop-v<version>/data** directory and placed in the present working directory or in **$HOME/.schrodinger/qikprop**.

By default the molecules are flagged as being dissimilar to other known drugs based on comparison to a set of test structures studied by Professor William L. Jorgensen. If you desire to use limits other than the default limits, you can edit *QPlimits*.

In the *QPlimits* file beneath the name of each property are the current limits for that property. These limits can be deleted and replaced by user-specified minimum and maximum values. Structures that have properties which do not lie within the limits will then be flagged with an asterisk in the output files.

To run in fast processing mode, the word `normal` at the end of the *QPlimits* file should be replaced by the word `fast`. An excerpt from *QPlimits* is shown below.

```
Molecular Weight                    standard is:  ( 130.0 / 725.0)
 130.0 / 725.0
Dipole Moment (D)                   standard is:  (   1.0 /  12.5)
   1.0 /  12.5
Total        SASA                   standard is:  ( 300.0 /1000.0)
 300.0 /1000.0
...
Mode (normal or fast)
normal
```

6.5 Specifying an Input Source

When running QikProp jobs from the Maestro interface, you can use Workspace structures or selected entries in the Project Table panel as input. To evaluate structures from a file, you must import the structures into a project using the Import panel, select the desired structures in the Project Table (all new structures are selected by default), and enable the Selected Entries button, located under Source of Job Input: on the QikProp panel. The default setting for QikProp job

input is Workspace. To perform QikProp evaluations for the structures currently appearing in the Workspace, you do not need to change the Source Of Job Input: setting.

6.6 Selecting a Job Output Incorporation Scheme

QikProp job output can be incorporated into the Project Table in several different ways. Select the desired incorporation scheme by selecting the appropriate Incorporate Output Into Project By: setting on the QikProp panel. By default, the option Replacing Existing Entries is selected. This option causes Maestro to replace each processed entry with the QikProp output structure and all of the predicted properties associated with that structure. Usually this amounts to the addition of 36 new columns in the Project Table and the import of 36 QikProp properties. Generally the output structure does not differ from the structure used as input. However, if QikProp determines that a structure is missing hydrogens, QikProp modifies the input structure and replaces it with the corrected structure.

When preparing QikProp calculations, you can also choose the Appending New Entries option, which leaves the original entries untouched and adds rows in the Project Table that contain property data for each of the processed structures. Appended rows appear at the end of the Project Table.

The third option allows you to choose not to incorporate the data into the project at all. If you select the Ignoring Output option, the output files are written to the specified directory, but the predicted properties do not appear in the Project Table. For more information on output files, see Chapter 7: “QikProp Output” on page 47.

6.7 Starting a QikProp Job

You can launch a QikProp job from Maestro by clicking the QikProp panel’s Start button. For more information on running QikProp from the command line, see Chapter 8: “Command Line QikProp” on page 51.

6.8 Monitoring the Job

After you click the Start button of the QikProp panel, the Monitor panel is displayed automatically, and the contents of the <*jobname*>.*out* file are displayed as the file is written.

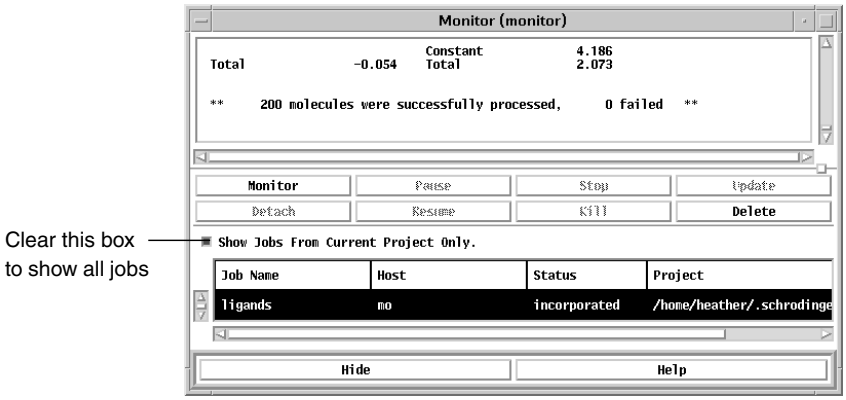

Figure 6.3. The Monitor ***panel***

The name of the job will be placed in the job database portion of the Monitor panel, and the job will be automatically selected in the database. By default, the Show Jobs From Current Project Only option is selected. To show all jobs run from your user name, clear the check box.

When there are no jobs currently running, only the Monitor and Delete buttons are active. While a job is running, the Detach, Pause, Resume, Stop, Kill, and Update buttons are active, but you can only use the Kill button. To terminate a QikProp job, select the desired job in the database list, and then click the Kill button.

Chapter 7: QikProp Output

7.1 Output File Types

For every completed QikProp job, the files:

<jobname>-out.mae	*<jobname>.CSV*
<jobname>.mae	*<jobname>.qpsa*
<jobname>.out	

are created, where <jobname> is the stem of the input file name. For example, a job with an input file named "*ligs.mae*" produces output files named *ligs-out.mae*, *ligs.mae*, and so on. In addition to these five standard files, a *<jobname>.warning* file is written if QikProp has difficulty processing a structure.

7.1.1 The <jobname>-out.mae File

Upon completion of a job, QikProp creates the *<jobname>-out.mae* file. This is a Maestro-formatted structure file that contains the QikProp properties generated for each structure as well as structural information. Importing the *<jobname>-out.mae* file into any future Maestro session places each of the structures in the file, along with the associated properties, in a separate entry in the Project Table.

7.1.2 The <jobname>.mae File

Before Maestro passes a QikProp job on to the Job Control Facility, QikProp first writes a Maestro-formatted *<jobname>.mae* file containing structural information for the molecules to be evaluated. The structure contained in the *<jobname>.mae* file depends on the Source of Job Input: setting. If you selected Workspace in the QikProp panel, the *<jobname>.mae* file contains any structures that appeared in the Workspace when the job was started. If you selected Selected Entries, the file contains the structures corresponding to the entries in the Project Table that were selected when the job was started.

7.1.3 The <jobname>.qpsa File

The <*jobname*>.*qpsa* file contains additional information for each molecule, including the covalent neighbors list, atom-type assignments, associated atom-type force field parameters, and a solvent accessible surface area (SASA) analysis. At the end of the file, there is a summary giving the number of molecules that were successfully processed and the number of molecules that were not successfully processed, and a report of the processor time used.

7.1.4 The <jobname>.out File

The <*jobname*>.*out* file lists the output descriptors and predicted properties. In addition, the file contains results from comparisons between the compound's properties and the properties of typical drugs. Each property with a value outside the 95% range for known drugs is marked with an asterisk (*).

Also in the <*jobname*>.*out* file is a "QP Breakdown" section for each structure. The four quantities listed in the breakdown, log Po/w, -log S, log BB and log PMDCK, are computed as a linear combination of more basic molecular properties. In the breakdown, QikProp lists the descriptor values multiplied by their coefficients for each derived property which, if added, yields the value listed under `Total`, which is the predicted property value. Descriptor values that are above the maximum observed in the training set used to develop QikProp are marked with an angle bracket (<).

7.1.5 The <jobname>.CSV File

The <*jobname*>.*CSV* file is a comma-separated-value (CSV) file that lists the 36 QikProp descriptors and predicted properties on one comma-separated line per molecule. The Unix version of QikProp creates a <*jobname*>.*CSV* file for each job. As with the other output files, the stem of the <*jobname*>.*CSV* file corresponds to the jobname entered in the Maestro QikProp panel.

The format of the <*jobname*>.*CSV* file allows you to import it into spreadsheet programs such as Excel, or analysis programs such as JMP. You might want to do this to create a training file for use with QikFit. See 9.3 "The Training File" on page 57 for more information about QikFit training files. When QikProp writes this file, it converts any commas appearing in molecule names to semicolons, and it removes any blanks from molecule names. The <*jobname*>.*CSV* file can also be used as input for QikSim jobs.

QikProp makes entries in the <*jobname*>.*CSV* file for any molecules with fatal errors. Check the file for these messages after multiple-structure runs. See 7.1.6 “The <jobname>.warning File” on page 49 for more information about warnings.

7.1.6 The <jobname>.warning File

Warnings about potential problems with the input structures are written to the <*jobname*>.*warning* file. For instance, warnings are issued when QikProp encounters empty input files, ionic input structures, structures with multiple fragments, and structures with missing hydrogens.

In addition to statements placed in the <*jobname*>.*warning* file, if multiple structures are being processed in a single run, any structures with fatal errors (e.g., disallowed atom types) are normally skipped over, and entries are made in the <*jobname*>.*CSV* file stating that the processing of the molecules failed. After executing QikProp for multiple structures, you should (a) display the contents of the <*jobname*>.*warning* file, if present, and (b) search for any occurrences of `empty` or `failed` in the <*jobname*>.*CSV* file. If QikProp encounters no problems with the processed structure(s), a <*jobname*>.*warning* file is not generated.

7.2 Output File Directory

QikProp places any generated output files in the file i/o directory specified in the Directory folder of Maestro’s Preferences panel.

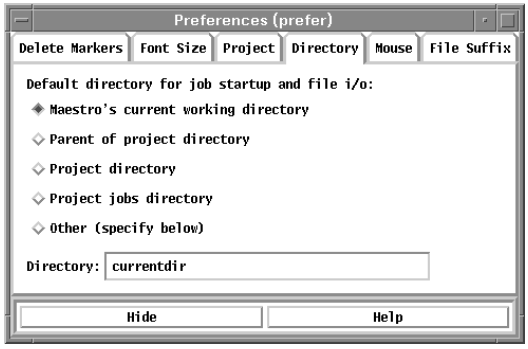

Figure 7.1. The Directory ***folder of the*** Preferences ***panel***

The default file i/o directory is Maestro's current working directory. To change the file i/o directory preference, open the Preferences panel from the Maestro menu in the Maestro Main Application Window. Then click the toggle button corresponding to the desired directory option. Click Hide to close the panel.

Note that the default file i/o directory, as well as most of the other directory Preference choices, is relative. For instance, if you select the Maestro's current working directory option, Maestro will write output files to the directory from which you launched Maestro, unless you have since entered the `cd` or `chdir` command in the Command Input Area.

Chapter 8: Command Line QikProp

8.1 Introduction

QikProp can be run from the command line as well as from the Maestro user interface. You might want to run QikProp from the command line in the following situations:

- When you want to simultaneously process more than 500 structures.
- When you are processing structures but are postponing structure viewing and analysis.

8.2 Issuing the 'qikprop' Command

To use QikProp from the command line, open a terminal window and set the SCHRODINGER environment variable, if it has not been set already. For more information, see section 4.2 “Environment Variables” on page 15.

To run QikProp, enter the command:

```
$SCHRODINGER/qikprop [<options>] <input_file>
```

replacing [<options>] with any desired command line options, and replacing <input_file> with the name (including path if the file is not in your current working directory) of the file to be evaluated. See section 8.3 “Input Files” below for more information on acceptable input file types.

The QikProp command line options are described in the table below.

`-HOST <host>`	Run the job on the specified remote host.
`-LOCAL`	Use the current working directory rather than a temporary scratch directory for scratch files.
`-WAIT`	Do not return to the command prompt until the job finishes. The default is to run the job in the background.
`-NICE`	Run the job at reduced priority.
`-HELP`	Print the acceptable command line arguments and exit.

QikProp jobs, regardless of whether they are submitted from the Maestro GUI or from the command line, are managed by the Schrödinger Job Control Facility. All jobs, whether started from the command line or from Maestro, can be monitored from Maestro's Monitor panel. For information on the Job Control Facility and how to use the Facility to monitor QikProp jobs, see Appendix B: "Command Line Job Control" on page 75.

8.3 Input Files

QikProp input must be a file containing the 3D structure (x,y,z coordinates and atomic numbers) of one or more molecules. Four file formats are accepted:

- Maestro-formatted *.mae* files
- MDL mol files, also known as 3D SD files
- PDB files
- BOSS/MCPRO Z-matrix files

An example of each file type is provided in the **$SCHRODINGER/qikprop-v<version>/molfiles** directory. In addition, this subdirectory contains the mol files for 30 common drugs. See section D.1 "Sample mol File" on page 93 for more information about the SD *.mol* file format.

QikProp input structures and files must meet the following criteria:

- Hydrogens on input structures must be explicit. You cannot, therefore, use as input united atom-typed files from molecular mechanics programs. If you attempt to use structures with the united atom hydrogen treatment, Maestro will prompt you to use an alternative hydrogen treatment. There is no such check for missing hydrogens when you run QikProp from the command line.
- With the exception of $$$$-delimited MDL SD mol files and Maestro-formatted files, each structure file must contain only one molecule. You can concatenate MDL SD mol (<*jobname*>*.mol*) files into a single file for input to QikProp. For an example, see the file *$SCHRODINGER/qikprop-v<version>/molfiles/many.mol*.
- Functional groups on input structures must be in their neutral state, e.g., amines cannot be protonated and acids cannot be deprotonated.

QikProp makes adjustments for the proper protonation state in its predictions.

- For mol and PDB files, the allowed atom types are H, C, N, O, F, Al, Si, P, S, Cl, Br, and I. In fast processing mode all atom types are permitted, though some results might not be valid if atoms other than the above PM3 atoms are used.

Chapter 9: QikFit

9.1 Introduction

QikFit generates linear regression equations that describe the relationship between a new, user-added experimental property and applicable default QikProp descriptors and properties. Using the following equation, QikFit performs iterative calculation cycles to obtain optimized coefficients for QikProp properties that correlate linearly with the new property:

$$P_j = \sum_i c_i \chi_{ij} + c_0$$

In the above equation, P_j is the property or activity that is to be predicted for molecule j, the c_i values are the regression coefficients, χ_{ij} is the ith relevant property for molecule j, and c_0 is a constant. At the end of each calculation cycle, QikFit discards properties that are not correlated with the new property by more than the significance cutoff value. This value is 0.0006 by default.

When exactly 37 descriptors and properties are present in the input file (e.g. *<jobname>.CSV* if you generated the input file with QikProp), QikFit generates a *<jobname>.myfits* file. QikProp can then use the values in this file to predict the new property for other similar molecules.

You can use the *<jobname>.myfits* file to integrate into QikProp coefficients from your own previous QSPR/QSAR analyses. This allows you to use those coefficients, rather than the provided default values, to calculate the standard 36 QikProp descriptors and properties.

In addition to the regression analysis described above, QikFit can also perform a trend vector analysis. This is generally used when there are a large number of descriptors and the ratio of molecules to descriptors is not high. For a regression analysis, there should be at least five times as many molecules as descriptors.

QikFit has not been interfaced with Maestro and therefore must be run from a terminal window. See section 9.2 below for more information about executing the program.

9.2 Running QikFit

To use QikFit from the command line, open a terminal window and set the SCHRODINGER environment variable, if it has not been set already. For more information, see section 4.2 “Environment Variables” on page 15.

To run QikFit, enter the command:

```
$SCHRODINGER/qikfit [<options>] <training_file>
```

Here, <*training-file*> is the name (including the path if the file is not in your current working directory) of the training file to be evaluated. The training file is a comma-separated-value (CSV) file containing the QikProp-calculated properties to be used in the regression, and a column of activities to be fit. See section 9.3 below for more information on training files.

The optional QikFit command line arguments are described in the table below. For more information, see the relevant sections later in this chapter.

`-a` <column#>	The column containing the new property data in the training file. A <column#> of `0` selects the last column. By default, the last column is assumed to hold the new property data.
`-c` <cutoff>	Properties whose contribution to the fit are below the significance cutoff are iteratively removed from the fit. The default cutoff is 0.0006.
`-o` <out-file>	The file in which the output of the analysis is written. The default is <*jobname*>*.out*.
`-t` <test-file>	An optional comma-separated-value (CSV) file containing the QikProp properties of molecules for which activity values will be predicted using the best-fit coefficients.
`-x` <column#>	A column to be excluded from the regression analysis. By default, all columns are always used. Only one column to be excluded can be specified. To exclude more than one column, you must edit the file and delete the columns.
`-z`	Perform a trend-vector analysis, rather than the standard regression analysis.

-HOST <host>	Run the job on the specified remote host.
-WAIT	Run the job in the foreground (do not return to the command prompt until the job ends). The default is to run the job in the background (return to the command prompt immediately).

9.3 The Training File

To use QikFit to predict an additional property, you must have experimental data for enough compounds to comprise a "training set" with which regression equations can be generated. For a regression analysis, it is recommended that you have five times more training compounds than descriptors (approximately 175 compounds). If you have fewer compounds than this, it is recommended that you perform a trend vector analysis rather than a regression analysis.

The training file must contain molecular properties and descriptors, as well as the new experimental property values for all of the structures.

Requirements for the Training File Format:

- The training file must be in CSV format. If you are using as a training file <*jobname*>.*CSV*, generated by QikProp, this file is already in CSV format. To this file you must add a column containing the new experimental property values.
- The first line must be a header with names for the columns.
- The second and subsequent lines must contain a molecule name or code in the first column, followed by the molecular properties and descriptors.
- The new experimental property should be placed in the last column of each line. If you place it in a different column, you must specify the column number as a command line argument. Use a spreadsheet program to add the data column.
- The number of descriptors that can be used for regression analyses is limited to 250.
- For trend vector analyses, 2000 descriptors are allowed.

- For an analysis of either type, no more than 2500 molecules can be processed in the same file.
- The length of each line is limited to 10,000 characters.

9.4 The Test File

You can specify the name of a test file as a command line argument when you run QikFit. This test file should contain descriptors and properties for molecules that are different from, but structurally similar to, those in the training set. If you specify a test file, the optimized regression calculated for the training set is used to predict the new experimental property for the molecules in this file.

Format Restrictions for the Test File

- The test file must be in CSV format.
- The test file must not have a column containing the new experimental property.
- The test file must have the same number of descriptors and properties as the training file, not including the new experimental property.
- The descriptors and properties must be in the same order as in the training file. This order is the order in which they would appear if the file had been generated using QikProp. (It is possible to generate the test and training files with programs other than QikProp so long as the QikProp property order is preserved.)

9.5 Significance Cutoff Values

For the regression analysis only, descriptors with little effect on the fit are pruned in a series of up to 10 calculation cycles. The significance is a measure of the quantitative effect of a change in the descriptor on the sum of squares error (SSE). Selecting higher values for the significance will cause more descriptors to be discarded. The default significance cutoff is 0.0006. If you want to change this value, use the `-c` option to specify an alternative value.

9.6 Running on a Remote Host

Before you can run a QikFit job on a remote host, you must configure your Schrödinger Job Control facility by adding the host names that you want to use to the *schrodinger.hosts* file. For more information on Job Control configuration, see Appendix A: "Configuring Job Control" on page 69.

After you have Job Control configured, you can run QikFit jobs on a remote host by using the `-HOST` option to specify the name of the desired host.

9.7 Output Files

QikFit results are written to the *<jobname>.out* file. If QikFit finds exactly 37 descriptors and properties (36 from QikProp and one experimental value that you have added) in the training file, it generates a *<jobnname>.myfits* file. This file can be used with QikProp to predict the new property for additional, structurally similar molecules. In addition, a *<jobname>.log* file is produced. This file contains basic information about the job, such as what options were specified.

9.7.1 The <jobname>.out File

The *<jobname>.out* file contains results from the iterative cycles QikFit performs to determine which descriptors in the input file have linear relationships with the new property. The last line of the file contains the predicted new property for the structures in the test file, if one was provided.

Excerpts from the file:

$SCHRODINGER/qikprop-v<version>/tutorial/qikfit/ligands.out

are provided for both a regression analysis and a trend vector analysis. See Appendix C: "Sample Output Files" on page 81 for the excerpts.

9.7.2 The <jobname>.myfits File

This file contains coefficients which, when used with QikProp descriptors and properties, can be used to generate the new experimental property for molecules that are structurally similar to those in the training set.

A <*jobname*>.*myfits* file is only written during a QikFit run when there are exactly 37 descriptors and properties (36 default QikProp descriptors and properties and the additional experimental activity column you have added) in the <*jobname*>.*CSV* training file.

QikProp does not attempt to determine the identity of the data in the <*jobnname*>.*myfits* file. It is therefore necessary that the 36 default QikProp descriptors and properties appear in the <*jobname*>.*myfits* file, and that they appear in the same order in which QikProp generates them.

The <*jobname*>.*myfits* file, for each of 4 possible fits, lists the 36 QikProp-predicted descriptors, properties, and coefficients that relate these values to the new experimental property.

9.8 Using QikFit Output To Predict New Properties

You can use the coefficients file <*jobname*>.*myfits* with QikProp to predict additional properties, using the following procedure.

1. Open the <*jobname*>.*myfits* file in a text editor.
2. Above the title for the first fit, enter the number of fits you want QikProp to read from the file. For instance, if you have added data for only one new property, enter `1`.
3. Below the title for each fit, enter the name of the property you would like to predict. This name appears when you print out the file, and it also appears as the column heading in the <*jobname*>.*CSV* file.
4. Save the file as *QPmyfits* into the current working directory.
5. Run QikProp

See “Predicting New Properties with the QPmyfits File” on page 33 for an example of how to predict properties. See Appendix C: “Sample Output Files” on page 81 for a sample portion of the *$SCHRODINGER/qikprop-v<version>/data/myfits* file.

Chapter 10: QikSim

10.1 Introduction

QikSim performs a similarity/diversity analysis for a designated molecule in a QikProp <*jobname*>.*CSV* file. During the analysis, QikSim compares the molecule to all the other molecules in that <*jobname*>.*CSV* file by first computing the Euclidean distance and the Tanimoto coefficient that relate the probe molecule to the others. QikSim then sorts results from most similar to least similar. Such analyses are useful for deriving new leads from a given reference compound.

10.2 Command Syntax

To run QikSim, enter the following command from a terminal window in which the SCHRODINGER environment variable has been set:

```
$SCHRODINGER/qiksim [<options>] <CSV_file>
```

Here, <CSV_file> is the name of the file to be evaluated. This file is a comma-separated-values (CSV) file containing the QikProp-calculated properties of the molecules to be compared.

The optional QikSim command line arguments are described in the table below.

`-o` <out_file>	The file in which the output of the QikSim analysis is written. The default is <*jobname*>.*out*.
`-l` <limits_file>	A file containing the range in which specified property values for each comparison molecule must lie. By default, the *limits.def* file is used.
`-p` <molecule#>	The row in the <CSV_file> that contains the probe molecule. The default is `0`, which selects the last row.
`-n` <nkeep>	The number of most-similar molecules to report. By default, all are reported.
`-HOST` <host>	Run the job on the specified remote host.
`-WAIT`	Do not return to the command prompt until the job finishes. The default is to run the job in the background.

10.3 Input

A QikSim analysis requires three input elements:

1. **A *<jobname>.CSV* file written by version 2.0 of QikProp.** A sample *CSV* file is provided in the directory:

   ```
   $SCHRODINGER/qiksim-v1.0/testcases
   ```

   This file contains QikProp 2.0 results for 555 molecules, most of which are known drugs.

2. **A *limits* file containing the limits and weights for each QikProp descriptor that is used for the QikSim similarity/diversity analysis.** A molecule is output as similar or dissimilar if each of its QikProp descriptors falls between the listed minimum and maximum values. Otherwise the molecule does not appear in the output. The weights can be used linearly to emphasize or de-emphasize the contribution of a descriptor to the similarity/dissimilarity measure. The file begins with:

```
Descriptor/min, max, weight:
#Stars      Number of stars
    0 1000  0.0
#Amines     Number of non-conjugated amines
    0 1000  0.0
#Nitro      Number of nitro groups
    0 1000  0.0
#Acid       Number of carboxylic acid groups
    0 1000  4.0
#Amides     Number of non-conjugated amides
    0 1000  0.0
#Rotor     Number of easily rotatable bonds except for CXYZ (X,Y,Z
= H, halogen) groups
    0 1000  1.0
etc.
```

The Euclidian distance calculation uses the weights from the *limits* file. The Tanimoto coefficient calculation does not use the values of the weights, but descriptors are ignored if they have zero weight.

The *limits* file can be specified on the QikSim command line. If no file is explicitly given, the program looks for a file called *limits* in the current directory. If no such file exists, the default *limits* file, *$SCHRODINGER/qiksim-v1.0/data/limits.def*, is used. If you want to use a customized *limits* file, you should copy the default file to your working directory and modify it to suit your purposes.

3. **The row number of the probe molecule in the CSV file and the number of results to report.** You must look at the *CSV* file to find the row number of the probe molecule. Remember that molecule x is in row x+1 owing to the header in line 1; for instance, molecule 392 is on line 393 of the *CSV* file. The default row is the last row.

Chapter 11: Bibliography

11.1 QikProp

Duffy, E. M.; Jorgensen, W. L. Prediction of Properties from Simulations: Free Energies of Solvation in Hexadecane, Octanol, and Water. *J. Am. Chem. Soc*. **2000**, 122, 2878-2888.

Jorgensen, W. L.; Duffy, E. M. Prediction of Drug Solubility from Monte Carlo Simulations. *Bioorg. Med. Chem. Lett*. **2000**, 10, 1155-1158.

Jorgensen, W. L.; Duffy, E. M. Prediction of Drug Solubility from Structure, *Advanced Drug Delivery Reviews,* (2002), in press.

Jorgensen, W. L.; Duffy, E. M.; et al. Prediction of Cell Permeability from Simulations. To be published.

Jorgensen, W. L. Prediction of Blood-Brain Barrier Penetration and CNS Activity. To be published.

Colmenarejo et al. Cheminformatic Models To Predict Binding Affinities to Human Serum Albumin. *J. Med. Chem.* **2001**, 44, 4370-4378.

11.2 QikFit

Multiple Linear Regression:

Kleinbaum, D. G.; Kupper, L. L.; Muller, K. E.; Nizam, A. Applied Regression Analysis and Other Multivariable Methods. Third Edition, Duxbury Press: Pacific Grove, 1987.

Trend Vector:

Carhart, R. E.; Smith, D. H.; Venkataraghavan, R. *J. Chem. Inf. Comput. Sci.* **1985**, *25*, 64-73.

Chapter 12: Help

12.1 Getting Help on Program Installation

For help with installing Maestro, QikProp, QikFit, and QikSim, see the *Schrödinger Product Installation Guide* included with your shipment.

12.2 Getting Help from within Maestro

The Maestro help facility consists of Balloon Help, Auto-Help, and online help. To get help on operating Maestro, follow the steps below.

- Check the Auto-Help window, located below the title bar of the Main Application Window. If Auto-Help is available for the task you are performing, it will automatically be displayed there.
- If the question you have concerns an interface element, e.g., a button or option menu, there may be Balloon Help for the item. Position the mouse pointer over the element and pause in that position. If there is Balloon Help for the element, it appears within a few seconds in a gold text box.
- If you have not found the help you need using either of the steps above, click the Help button in the lower right corner of the panel for whose settings you are seeking help. The Help panel will be opened, and a relevant help topic will be displayed.
- If you are seeking help for a concept or action not associated with a panel, you can access the Help panel from the Help pull-down menu on the Main MenuBar. You can also display the Help panel from the Main Application Window using the key combination Alt+H.
- See the *Maestro User Manual*.

12.3 QikProp Help from the Command Line

To get help for the `qikprop`, `qikfit`, or `qiksim` commands, type $SCHRODINGER/<command_name> in a terminal window in which the SCHRODINGER environment variable has been set. Replace <command_name> with the command for which you want help.

12.4 Technical Support

If you have questions, contact Schrödinger using the information below. Generally, e-mail correspondence is best because you can send relevant machine output.

Schrödinger
E-mail: **help@schrodinger.com**
USPS: 1500 SW First Ave., Suite 1180, Portland, OR 97201
Phone: (503) 299-1150
Fax: (503) 299-4532
WWW: http://www.schrodinger.com
FTP: ftp://ftp.schrodinger.com

When sending e-mail messages, please include the following:

- all relevant user input and machine output
- QikProp purchaser (company, research institution, or individual)
- primary QikProp user
- computer platform type
- operating system with version number
- QikProp version number
- Maestro version number
- mmshare version number

Much of the machine and system information listed above can be produced by entering the command:

```
$SCHRODINGER/machid
```

at a shell prompt once the SCHRODINGER environment variable has been set.

Appendix A: Configuring Job Control

A.1 Introduction

For an overview of the `jobcontrol` command line utility, see Appendix B: "Command Line Job Control" on page 75.

Configuration of the Job Control Facility is done in the *schrodinger.hosts* file. The default installation of this file is configured to allow running of jobs locally, and no action is required by the user. However, if you want to run distributed jobs, run jobs remotely, or change the location of the scratch directory (where job files are placed temporarily while the job runs), then Job Control must be configured. The following sections describe how to specify the scratch directory, how to prepare your account for remote job submissions, and how to modify the *schrodinger.hosts* file.

A.2 The Location of the Scratch Directory

When a scratch directory is created automatically, it is named:

<tmpdir>/<userid>/<unique_dir_name>

The <unique_dir_name> is usually the job name, but it could also have a sequence number appended to it in order to make the directory name unique.

There are a number of ways to specify the **<tmpdir>** directory. The Job Control system uses the first specification found from the following list:

- The directory specified on the command line using the `-TMPDIR` option. For example, to specify the scratch directory for QikProp:

```
$SCHRODINGER/qikprop -TMPDIR /scr/qp_tmp <input_file>
```

- The directory specified by the environment variable SCHRODINGER_TMPDIR, if this variable is set on the machine on which the job is started. For example, a csh or tcsh user might enter the commands:

```
setenv SCHRODINGER_TMPDIR /scr/qp_tmp
```

```
$SCHRODINGER/qikprop <job_name>
```

- If there is a `'tmpdir:'` line for the host entry in the host database file, that location is used for **<tmpdir>**.

For remote jobs, the following are also considered, if none of the above specifications is found:

- The directory specified by the environment variable SCHRODINGER_TMPDIR on the remote machine.
- The location indicated by the environment variable TMPDIR on the remote machine.

Finally, if no other specification for **<tmpdir>** is found, the directory **$HOME/.schrodinger/tmp** is used. Environment variables on the remote machine are created by invoking the user's remote shell ("rsh" or "ssh") in the normal manner. In all cases, **<tmpdir>** is created if it does not exist.

When the job is finished, the scratch directory is automatically removed if the following conditions are met:

- The job's output files were all successfully copied back to the startup directory.
- The scratch directory didn't exist before the job started.

A.3 Preparing for Remote Job Submission

To run remote or distributed jobs, it must be possible to execute remote commands using the '`rsh`' command or the '`ssh`' command without specifying a password.

This typically requires that either:

i) A *hosts.equiv* file exists in the **/etc** directory on each of the computers. This file should contain a list of computers amongst which users can log in without giving their passwords (provided that their userids are the same on each of the machines). Creating a *hosts.equiv* file usually requires root permission.

or

ii) A *.rhosts* file exists in the user's home directory on each of the remote computers. The *.rhosts* file should list the names of the computers and user name from which the user would like to login without specifying a password. The list should contain two lines for each machine—one with the machine name alone and one with the fully qualified name. Your *.rhosts* file should contain entries in the form of:

```
<machine1> <your_userid_on_machine1>
<machine1.your_dept.domain> <your_userid_on_machine1>
```

The userid in the *.rhosts* file is optional if the userid is the same on the remote hosts. You do not need root permission to configure this file, but you must make sure that the file does not have group write permission. Once you have configured your *hosts.equiv* or *.rhosts* file, use the following command to check for successful communication between the computer that the job will be started on and each of the other computers that the job will use.

```
rsh -l <your_userid_on_machine1> <machine1> date
```

This command should print the date from the computer <machine1>.

If you have a *schrodinger.hosts* file, you can automatically check all of the machines listed in it (see A.4.3 "Testing the schrodinger.hosts File" on page 73).

A.4 The schrodinger.hosts File

The Job Control Facility obtains information about the computers (hosts) on which it will run jobs from the *schrodinger.hosts* file.

A.4.1 Structure of the schrodinger.hosts File

A sample *schrodinger.hosts* file is shown below. The lines beginning with a # are comments and not required.

```
# Schrodinger hosts file
#
name:         localhost
schrodinger: /usr/local/schrodinger_72b
#
name:         ahost
user:         your_userid_on_ahost
processors:   1
tmpdir:       /scr
#
name:         big_host
user:         your_userid_on_big_host
processors:   16
tmpdir:       /big_scr
schrodinger: /software/Schrodinger_mmod72
#
name:         another_host
processors:   2
tmpdir:       /scr
schrodinger: /software/Schrodinger_mmod72
#
# End of Schrodinger hosts file
```

For each computer that will be used, there should be an entry in the *schrodinger.hosts* file specifying the name of the computer. In the example above, the name of the first computer is `ahost`. Additional lines can be specified for each host. There can also be a special host entry with the name `localhost`, which represents the computer that the user is logged into, regardless of what the computer is called.

The `user` line specifies the userid that will be used to log in to the remote host. If the userid is the same on the job startup computer as on the remote host, the userid for that remote host does not need to be specified in the *schrodinger.hosts* file.

The `processors` line indicates the maximum number of processors that a parallel or distributed job should use on that computer. This defaults to `1` if it is not given.

The `tmpdir` line for a given host specifies the location used for temporary files on that host. This directory should be local to the remote machine, e.g. not NFS-mounted, it should have sufficient disk space available, and the user should have write access.

The `schrodinger` entries are used to specify the location of the directory in which your Schrödinger software is installed on the remote machine. This entry allows the user to select which of several versions of our software to use on the remote machine, in case more than one version is installed. The `schrodinger` entry is also useful if a user's default shell does not set the SCHRODINGER environment variable on remote machines when using rsh or ssh.

A.4.2 Selection of the schrodinger.hosts File

The hosts database file used for a given job is the first one found in the following list:

1. The file specified by the (optional) environment variable SCHRODINGER_HOSTS
2. The *schrodinger.hosts* file in the current directory
3. The *schrodinger.hosts* file in **$HOME/.schrodinger**
4. The *schrodinger.hosts* file in **$SCHRODINGER**

A.4.3 Testing the schrodinger.hosts File

The *schrodinger.hosts* file can be tested using the command:

```
$SCHRODINGER/hunt -rtest
```

This command tests the first *schrodinger.hosts* file found in the list above, excluding the one specified by `-HOSTFILE`. This command also attempts to contact each of the hosts listed, and thus also serves as a check on host accessibility and on whether remote login has been set up appropriately on the various hosts.

Appendix B: Command Line Job Control

B.1 Introduction

The *$SCHRODINGER/jobcontrol* utility allows you to perform a number of job control tasks from the command line. You can perform many of these tasks using the Maestro Monitor panel as well, but some things (such as purging the jobs database) can only be done using the *jobcontrol* utility.

B.2 Jobcontrol Syntax and Supported Commands

The syntax for *jobcontrol* utility commands is:

```
$SCHRODINGER/jobcontrol <command> <job_selection>
```

where `<command>` is the command for the operation you want to perform, and `<job_selection>` is the name of the specific job, if applicable. The command can be one of the following:

Command	Action
`-list`	list the jobId, job name and status
`-show`	show the complete job record
`-kill`	terminate the job immediately
`-stop`	ask the job to stop itself as soon as possible (bmin only)
`-pause`	suspend the job temporarily
`-resume`	continue running a paused job
`-monitor <n>`	ask for monitoring files to be sent every <n> seconds
`-cancel`	cancel a job that has been launched, but not started
`-purge`	remove a completed job from the database
`-help`	describes the jobcontrol commands

For example, to list all currently running jobs, you would enter the command:

```
$SCHRODINGER/jobcontrol -list
```

The optional <job-selection> consists of one or more JobIds, job names, status codes or queries. It can also be the word all, to select all the jobs in the jobs database. The default selection is the query 'status!=completed'.

A query takes the form <field><op><value>, where <field> is one of the field names in the job record. The most useful fields for queries are: 'Name', 'Program', 'Host', 'Dir', 'JobHost', 'JobDir', 'Status' and 'ExitStatus'. The <op> is either '=' (equals) or '!=' (is not equal to). Use the wildcard character '?' to match a single unspecified character, or '*' to match zero or more unspecified characters. Examples of valid queries include 'program=impact', 'jobhost=abigail', 'exitstatus!=finished', 'name=*tmp', 'dir=*/my-project-dir'.

B.3 Using jobcontrol to Purge the Job Database

Normally, when running jobs from Maestro's Project Facility, the database record for a job will be cleared once the job is incorporated back into a Maestro project. Not all jobs are incorporable, however; for this and other reasons, job records might not get deleted automatically and the **.jobdb** directory might become full over time. The *$SCHRODINGER/jobcontrol* utility can be used to purge the job database. For example, the following command purges the entire database:

```
$SCHRODINGER/jobcontrol -purge all
```

This command only affects completed jobs. Running jobs cannot be purged. You must be careful not to purge completed jobs whose output you still intend to incorporate back into a Maestro project.

B.4 Default Behaviors

All jobs run in the background automatically: they are disassociated from the terminal session (or application) from which they were launched. As a result, the job continues to run even if you quit Maestro or the terminal session from which you launched the job. For command line jobs, the UNIX command prompt returns right away, without waiting for the job to finish. This behavior isn't always desirable, especially if you want to run the job in a script, in which some subsequent action can be taken only after the job finishes. To accommodate use in a script, the command line option -WAIT can be used to prevent the command prompt from returning until after the job finishes. For example:

```
$SCHRODINGER/qikprop -WAIT <job_name>
```

Even with this option, however, the calculation is still placed in the background, so pressing Ctrl+C or Ctrl+Z will not affect it.

Every job is assigned a unique JobId, consisting of the name of the machine on which the job was launched, a sequence number, and a hexadecimal timestamp, e.g., `'mirabelle-0-a1b2c3d4'`.

B.5 Remote Job Submission

The Job Control Facility allows jobs to be run on a remote host. When remote job submission is possible, the name of the remote machine is selected using a popup menu in the Maestro panel from which the job is launched. The special host name "localhost" can always be used to launch a job on the current machine.

A MacroModel, Impact, or QikProp command line job can be submitted to a remote host by using the `'-HOST'` option to specify the remote host name. For instance:

```
$SCHRODINGER/qikprop -HOST <host_name> <job_name>
```

Certain information about the remote machine is required to launch a remote job. This information is taken from the *schrodinger.hosts* file, which is described in some detail in section A.4 "The schrodinger.hosts File" on page 71. It may be necessary for you or your system administrator to edit this file in order to make remote jobs possible. In particular, the machines listed in Maestro's popup host-selection menu are taken from the *schrodinger.hosts* file.

B.6 The Job Control Scratch Directory

Most jobs now run in a scratch directory by default, rather than in the directory from which the job was started. When a scratch directory is used, output and log files are created there and are only copied back to the output directory when the job finishes, or when the job is being monitored in Maestro.

When you run a job from the command line, some applications, for instance MacroModel, allow you to ask for a job to be run directly in the job's startup directory instead. The `'-LOCAL'` option is used for this purpose. For example:

```
$SCHRODINGER/qikprop -LOCAL <job_name>
```

Not every application supports this option. When it is supported, it can be used for both local and remote jobs. If you run a remote job with the `-LOCAL` option, it is important to make sure that the startup directory is accessible from the remote machine.

The location of the scratch directory is normally chosen automatically, according to settings made in the *schrodinger.hosts* file. However, there are a number of alternative ways to control where it will be created. For more information, see Appendix A.2 "The Location of the Scratch Directory" on page 69.

B.7 The Job Record

The *$SCHRODINGER/jobcontrol* utility can also be used to examine the data in a job's job record. The `'-show'` action lists the complete database record for one or more jobs. To make sense of this output, it is useful to know that the job record is just a list of fields, one on each line, each consisting of a field name and its value. Many of these fields contain information that is only useful to the Job Control Facility, but a number may also be useful to users. Some of the latter include:

Field name	Meaning
JobId	The job's JobId
ParentJobId	The JobId of the parent job, if this is a subjob
Name	The job name
Program	The program name
Project	The project name
JobDir	The directory in which the job is run
JobHost	The machine on which the job is run
JobUser	The userid under which the job is run

Host	The machine from which job was launched
User	The userid under which job was launched
Dir	The startup and output directory (on Host)
Status	The current job status
ExitStatus	The reason the calculation stopped
Command	The command used to start up the actual calculation
Env	Environment variable settings for the job
StatusTime	The time at which Status was last updated
StopTime	The time at which the calculation stopped
StartTime	The time at which the calculation started running
LaunchTime	The time at which the job was submitted
MonitorInterval	The interval (sec) between monitoring updates (0 if off)
StructureMonitorFile	The name of the monitoring file holding the molecular structure
StructureOutputFile	The name of the file holding the final molecular structure
InputFiles	Files copied from the output directory to the job directory at startup
OutputFiles	Files copied from the job directory to the output directory at exit
MonitorFiles	Files copied from the job directory to the output directory during monitoring
LogFiles	Monitoring files that grow throughout the job

Appendix C: Sample Output Files

C.1 Sample Regression Output and Commentary

The following is an example of a *jobname.out* file produced by performing a QikFit linear regression. In this example, we fit to the last descriptor instead of adding an experimental value for QikFit to fit to. Commentary on the file contents is provided in bold.

```
      QikFit 2.0 - Multiple Linear Regression Analysis

Incomplete entry - skipping molecule: "CMC00000004"
Incomplete entry - skipping molecule: "CMC00000047"
Incomplete entry - skipping molecule: "CMC00000152"
Incomplete entry - skipping molecule: "CMC00000160"
Incomplete entry - skipping molecule: "CMC00000163"
  Illegal Character in Descriptor: !
```

If an entry in the CSV file is incomplete (missing the activity or the value for a descriptor), or if the file contains an illegal character, the entry is ignored and a notice is printed at the top of the file.

```
Column      Contents      Column      Contents
   1       molecule          2       #stars
   3       #amine            4       #nitro
   5       #acid             6       #amide
   7       #rotor            8       #rctvFG
   9       CNS              10       MW
  11       dipole           12       SASA
  13       FOSA             14       FISA
  15       PISA             16       WPSA
  17       volume           18       donorHB
  19       accptHB          20       dip^2/V
  21       ACxDN^.5/SA      22       glob
  23       QPpolrz          24       QPlogPC16
  25       QPlogPoct        26       QPlogPw
  27       QPlogPo/w        28       QPlogS
  29       BIPCaco          30       AffyPCaco
  31       QPlogBB          32       AffyPMDCK
  33       QPlogKp          34       IP(eV)
  35       EA(eV)           36       #metabol
  37       QPlogKhsa        38       exp_data

Simplex exceeded maximum cycles: 9000
```

```
Regression did not converge in this cycle; not a problem - keep
going.
```

The results for the first cycle of the analysis are now output using all 37 descriptors. (The CSV file contains 38 columns, as listed above; however, column 1 contains the molecule name.) Standard statistical measures are output including the r^2 correlation coefficient, rms error, and Fischer F-statistic. A good background reference is: D. G. Kleinbaum, L. L. Kupper, K. E. Muller, and A. Nizam, "Applied Regression Analysis and Other Multivariable Methods, Third Edition", Duxbury Press: Pacific Grove, 1987.

```
 Regression Analysis - Cycle  1

          R-Squared         =      0.1061
          RMS Error         =      0.8173
          Average Error     =      0.5789
          Sum of Squares    =    132.2716
          F Statistic       =      0.5377
          Degrees of Freedom=          36

          No. of Data Points=         200
          Average Activity  =      0.9447
          Exptl Activ Range =     -2.8760 to      2.7062
          Calcd Activ Range =      0.1684 to      1.7288
          Activity in Column      38
          Signif. Cutoff    =      0.0006

 Descriptor     Parameter    Significance   Partial F    Partial R^2
  #stars         0.023571       0.0000        1.8153       0.0091
  #amine        -0.129837      -0.0001        0.0127       0.0001
  #nitro         0.000000       0.0000        0.0000       0.0000
  #acid         -0.147614       0.0001        0.3471       0.0018
  #amide         0.116963       0.0000        1.7936       0.0090
  #rotor         0.039570       0.0005        0.3740       0.0019
  #rctvFG       -0.219500       0.0001        0.9128       0.0046
  CNS            0.000056       0.0000        0.0444       0.0002
  MW            -0.003055       0.0094        4.5926       0.0227
  dipole        -0.002754       0.0000        2.5645       0.0128
  SASA          -0.001786       0.0105        5.4411       0.0267
  FOSA           0.000094       0.0000        2.9814       0.0148
  FISA           0.000335       0.0000        0.1149       0.0006
  PISA           0.000180       0.0001        0.0088       0.0000
  WPSA           0.001837       0.0002        0.2818       0.0014
  volume         0.000473       0.0026        6.1115       0.0299
  donorHB        0.049118       0.0001        0.2993       0.0015
  accptHB       -0.000908       0.0000        0.0001       0.0000
```

```
dip^2/V       -2.582746       0.0004        3.7342       0.0185
ACxDN^.5/SA  -13.058270       0.0006        1.2437       0.0062
glob          -0.082376       0.0000        3.2992       0.0164
QPpolrz        0.059584       0.0394        6.1028       0.0299
QPlogPC16      0.049495       0.0024        2.5284       0.0126
QPlogPoct     -0.055253       0.0101        0.9059       0.0046
QPlogPw       -0.002370       0.0000        0.1069       0.0005
QPlogPo/w     -0.138717       0.0011        4.2445       0.0210
QPlogS        -0.053854       0.0005        5.3725       0.0264
BIPCaco       -0.000005       0.0000        0.0001       0.0000
AffyPCaco     -0.000036       0.0003        0.1774       0.0009
QPlogBB       -0.019367       0.0000        0.5385       0.0027
AffyPMDCK     -0.000002       0.0000        0.0014       0.0000
QPlogKp       -0.144400       0.0041        0.2152       0.0011
IP(eV)        -0.005411       0.0000        0.0010       0.0000
EA(eV)         0.160889       0.0003        1.7278       0.0087
#metabol       0.025477       0.0001        0.2521       0.0013
QPlogKhsa      0.233016       0.0000        5.6344       0.0277
Constant       0.907894       0.0133
```

The next cycle is performed after removal of the descriptors with significance values below the cutoff (0.0006).

```
Regression Analysis - Cycle  2

         R-Squared         =      0.0776
         RMS Error         =      0.8303
         Average Error     =      0.5823
         Sum of Squares    =    136.4921
         F Statistic       =      1.7766
         Degrees of Freedom=           9

         No. of Data Points=         200
         Average Activity  =      0.9447
         Exptl Activ Range =     -2.8760 to      2.7062
         Calcd Activ Range =      0.4789 to      1.6891
         Activity in Column      38
         Signif. Cutoff    =      0.0006

Descriptor     Parameter    Significance   Partial F    Partial R^2
 MW            -0.001468       0.0022        4.5926       0.0227
 SASA          -0.003312       0.0380        5.4411       0.0267
 volume         0.001932       0.0374        6.1115       0.0299
 ACxDN^.5/SA   -2.233578       0.0000        1.2437       0.0062
 QPpolrz        0.051634       0.0276        6.1028       0.0299
 QPlogPC16     -0.181576       0.0265        2.5284       0.0126
 QPlogPoct      0.016400       0.0009        0.9059       0.0046
```

```
QPlogPo/w      0.142958       0.0013        4.2445       0.0210
QPlogKp       -0.160990       0.0048        0.2152       0.0011
Constant       0.378240       0.0021
```

The cycles continue until no more descriptors fall below the significance cut-off or until 10 cycles have been performed. The present case terminated on the third cycle. Out of the original 37 descriptors, 8 have been retained:

```
Regression Analysis - Cycle  3

         R-Squared         =      0.0775
         RMS Error         =      0.8303
         Average Error     =      0.5826
         Sum of Squares    =    136.5122
         F Statistic       =      2.0054
         Degrees of Freedom=           8

         No. of Data Points=         200
         Average Activity  =      0.9447
         Exptl Activ Range =     -2.8760 to      2.7062
         Calcd Activ Range =      0.4929 to      1.7123
         Activity in Column      38
         Signif. Cutoff    =      0.0006

Descriptor     Parameter    Significance   Partial F    Partial R^2
 MW            -0.001697       0.0030        4.5926       0.0227
 SASA          -0.003658       0.0464        5.4411       0.0267
 volume         0.002134       0.0456        6.1115       0.0299
 QPpolrz        0.052912       0.0290        6.1028       0.0299
 QPlogPC16     -0.182341       0.0267        2.5284       0.0126
 QPlogPoct      0.014890       0.0008        0.9059       0.0046
 QPlogPo/w      0.152722       0.0014        4.2445       0.0210
 QPlogKp       -0.161398       0.0048        0.2152       0.0011
 Constant       0.386967       0.0022
```

The provided experimental results and the results computed from the final regression equation are then listed for each molecule, and the final value for each QikProp parameter is listed. The values of parameters found not to have a significant linear relationship with the new experimental property are listed as '0.00000'.

```
   N  Exptl Activity Calculated    Delta      Compound
    1      1.331762         1.206324        0.125437        mol_0004
    2      1.217879         1.142674        0.075205        mol_0007
...etc.
    195    1.701800         1.127796        0.574004        mol_3013
    196    0.590873         0.914083       -0.323211        mol_3043
    197    1.545472         1.169439        0.376033        mol_3066
    198    0.332011         1.272036       -0.940025        mol_3067
    199    0.856038         1.145599       -0.289561        mol_3085
    200    1.838763         1.177887        0.660876        mol_3138

myfits file not written - need  37 descriptors.
```

The program notes that the QikProp *myfits* file has not been written in this case because the input CSV file did not contain all 37 QikProp descriptors and properties. This may be unintuitive at first, because there are 38 columns of data listed at the top of the file. Recall, however, that the first column contains the molecule name. Also, for this particular run, QikFit was told to fit to the last column of data, "QPlogKhsa." As a result, QikFit counts that column as experimental data and not as a QikProp descriptor, so to QikFit, it appears that one of the necessary descriptors is missing. In an actual QikFit run, you would add your experimental property to the 37 QikProp descriptors and properties, so there would be a total of 39 columns.

```
              New Predictions

    N    Calc Activity         Compound
    1     1.050474            "mol_7147"
```

Finally, predictions for any molecules in the test CSV file are then listed.

C.2 Trend Vector Output

Below is sample output from a trend vector analysis. Commentary is provided in boldface text. The output from the trend vector analysis is similar to output from a regression; however, there is only one analysis cycle and all descriptors are used. The following results are for the same data as above. This data also appears in the file called *out* produced by the QikFit run.

```
              QikFit 2.0 - Trend Vector Analysis


          R-Squared         =      0.8410
          RMS Error         =      0.2432
          Average Error     =      0.1977
          Sum of Squares    =      1.6562
          F Statistic       =     -0.9064
          Degrees of Freedom=          35

          No. of Data Points=          30
          Average Activity  =      0.1792
          Exptl Activ Range =     -0.9880 to      1.9110
          Calcd Activ Range =     -1.1742 to      1.3877
          Activity in Column      37
          T Vector Length   =        0.37


 Descriptor           Trend Vec
  #stars               0.072959
  #amine               0.042381
  #nitro               0.000000
  #acid                0.049569
  #amide               0.000000
  #rotor               0.003993
  #rctvFG              0.022023
  CNS                  0.053940
  MW                   0.079080
  dipole               0.000000
  SASA                 0.086363
  FOSA                 0.043515
  FISA                -0.084027
  PISA                 0.059661
  WPSA                 0.054163
  volume               0.086733
  donorHB             -0.077500
  accptHB             -0.057854
  dip^2/V              0.000000
  ACxDN^.5/SA         -0.082756
  glob                -0.075023
  QPpolrz              0.088957
  QPlogPC16            0.078093
  QPlogPoct            0.023344
  QPlogPw             -0.035054
  QPlogPo/w            0.113389
  QPlogS              -0.096070
  BIPCaco              0.052627
```

```
 AffyPCaco            0.045724
 QPlogBB              0.056059
 AffyPMDCK            0.062430
 QPlogKp              0.090000
 IP(eV)               0.000000
 EA(eV)               0.000000
 #metabol             0.067720
 Constant            -0.369492

N  Exptl Activity  Calculated      Delta         Compound
1    -0.218000     -0.060070     -0.157930    "acebutolol"
2    -0.988000     -1.174155      0.186155    "acyclovir"
3     0.385000      0.527218     -0.142218    "alprazolam(Xanax)"
4     0.060000      0.246985     -0.186985    "alprenolol"
5     0.047000      0.268094     -0.221094    "atropine"
6    -0.885000     -0.980918      0.095918    "AZT"
7     0.351000      0.132868      0.218132    "celecoxib"
8    -0.320000     -0.413756     0.093756    "cimetidine(Tagamet)"
9     0.328000      0.156298      0.171702    "delavirdineTot.E="
10    0.333000      0.424696     -0.091696    "fenfluramine"
11   -0.650000     -0.151212     -0.498788    "diflucan"
12    0.048000     -0.022045      0.070045    "ibuprofen"
13    0.090000      0.253660     -0.163660    "indomethacin"
14    0.802000      0.454939      0.347061    "lovastatin"
15    0.077000     -0.037306      0.114306    "lorazepam"
16    0.126000      0.168723     -0.042723    "MKC-442"
17    0.094000      0.292191     -0.198191    "nevirapine"
18    0.575000      0.385865      0.189135    "norvasc)"
19    0.090000     -0.159282      0.249282    "phenytoin"
20    0.660000      0.393375      0.266625    "progesterone"
21    0.665000      0.663545      0.001455    "prozac"
22    0.155000     -0.002271      0.157271    "rimadyl"
23   -0.450000     -0.000358     -0.449642    "rofecoxib"
24   -0.567000     -0.692126      0.125126    "serotonin"
25    0.310000      0.376020     -0.066020    "sustiva"
26    1.911000      1.387744      0.523256    "telmisartan"
27    0.972000      1.073706     -0.101706    "trifluprom azine"
28    0.694000      1.045378     -0.351378    "verapamil"
29   -0.248000      0.045336     -0.293336    "viagra"
30    0.930000      0.773856      0.156144    "zoloft(sertraline)"
```

The results of predictions for the input test CSV file, if any, are then listed. Note that the equation for trend vector predictions is:

$$P_j = \sum_i t_i \chi_{ij}^* + t_0$$

where the t_i are the trend vector coefficients, t_0 is the constant, and the χ* values are the normalized descriptors.

$$\chi_{ij}^* = (\chi_{ij} - \chi\,\min_i)/(\chi\,\max_i - \chi\,\min_i)$$

C.3 Sample myfits File

Below is an excerpt from a sample <*jobname*>.*myfits* file.

```
Use this file for your custom QSAR/QSPR fits to be output by Qik-
Prop 2.0. A maximum of 4 fits is allowed. For each fit, enter the
coefficient for each descriptor as a decimal number. Fit 1 below is
set-up as an example to be the same as the log Po/w equation used by
QikProp for non-hydrocarbons. Activate it by setting the number of
fits to 1. Enter the number of fits on the next line.
1
****************** Fit 1 *****************
Name for printing - 10 characters maximum
QPlogPo/w
constant
-0.6789
#amine
-0.5261
#nitro
0.0
#acid
0.0
#amide
-0.7265
#rotor
0.0
#rctvFG
0.0
CNS
```

```
0.0
MW
0.0
dipole
0.0
SASA
0.0
FOSA
0.0
FISA
0.0
PISA
0.0
WPSA
0.002753
volume
0.007269
donorHB
-0.5117
accptHB
-0.6552
dip^2/V
-4.512
ACxDN^.5/SA
67.42
glob
0.0
QPpolrz
0.0
QPlogPC16
0.0
QPlogPoct
0.0
QPlogPw
0.0
QPlogPo/w
0.0
QPlogS
0.0
BIPCaco
0.0
AffyPCaco
0.0
QPlogBB
0.0
AffyPMDCK
0.0
```

```
QPlogKp
0.0
IP(eV)
0.0
EA(eV)
0.0
```

C.4 QikSim Output

The sorted Euclidian distances and Tanimoto coefficients are listed in the file <*jobname*>*.simout*, following the input weights from the *limits* file. A typical example follows:

```
*********************************************************
*********************************************************
*************** Similarity/Diversity Analysis **************
*************** QikSim  Ver. 2.0 - June. 2002 **************
*********************************************************
*********************************************************

        Descriptor    Weight
        #Stars          0.00
        #Amines         1.00
        #Nitro          0.00
        #Acid           2.00
        #Amides         2.00
        #Rotor          1.00
        Reactv FGs      0.00
        CNS Actvty      0.00
        MW              1.00
        Dipole          1.00
        SASA            4.00
        FOSA            4.00
        FISA            4.00
        PISA            4.00
        WPSA            1.00
        Volume          4.00
        HB Donor        5.00
        HB Accptr       5.00
        Dip^2/V         1.00
        ADOSA           0.00
        Globularty      4.00
        Polariz.        1.00
        delG C16        1.00
        delG octnl      1.00
```

```
        delG water      1.00
        log P o/w       4.00
        log S           1.00
        PCaco-BI        1.00
        PCaco-Affy      1.00
        log BB          1.00
        PMDCK-Affy      1.00
        log Kp          0.00
        IP (eV)         0.00
        EA (eV)         0.00
        #Metabolic      0.00

        Number of Molecules =   555
```

Euclidean Distance			Tanimoto Coefficient		
N	R^2	Name	N	TC	Name
392	0.000	efavirenz(Sustiva)	392	1.000	efavirenz(Sustiva)
393	0.421	DPC961	393	0.995	DPC961
394	0.443	DPC963	394	0.993	DPC963
292	1.272	diuron	284	0.964	Diazepam(Valium)
383	1.306	neburon	295	0.958	DuPont697
401	1.631	thiocarcoxanilideUC781	373	0.950	midazolam
454	1.673	ralitoline93738-40-0	344	0.950	lansoprazole
379	1.780	nabumetone	401	0.932	thiocarcoxanilideUC781
552	1.961	chlorpyriphos	292	0.924	diuron
221	2.009	aminopyrine	356	0.917	loratidine
284	2.015	Diazepam(Valium)	402	0.917	pyridinoneL697661
386	2.030	2-methoxy-nevirapine(1	552	0.917	chlorpyriphos
430	2.030	Phenacetin	457	0.907	riluzole1744-22-5
402	2.032	pyridinoneL697661	304	0.904	etoricoxib
302	2.068	estrone	397	0.903	4-methylMBI
309	2.076	fenfluramine	387	0.898	N-methylnevanalog(12)
388	2.157	2-methyl-nevirapine	379	0.896	nabumetone
385	2.217	Nevirapine	405	0.895	dpc27pyridinoneNNRTI
396	2.383	8-chloro-TIBO(Tivirapine)	244	0.893	besipirdine119257-34
250	2.410	bupropion	221	0.891	aminopyrine
75	2.413	thymol	383	0.891	neburon
395	2.440	9-chloro-TIBO	504	0.884	trifluprom azine
417	2.441	Org12962	399	0.880	Trovirdine(PETT)
549	2.499	parathion	72	0.880	2;4;6-trichlorophenol
464	2.558	desmethylRTI55	168	0.878	EtEtOPOSEtSEt
300	2.578	estradiol	550	0.876	diazinon
295	2.618	DuPont697	71	0.868	2;3-dichlorophenol
73	2.650	p-t-butylphenol	75	0.864	thymol
471	2.695	sertraline	462	0.864	rotenone
234	2.706	atrazine	551	0.858	malathion

Appendix D: Sample Input Files

D.1 Sample mol File

QikProp complies with the MDL SD mol format specified by MDL in *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 244-255. Sample mol files are provided in the **$SCHRODINGER/qikprop/molfiles** directory. Below are the contents of the *flucon.mol* file in that directory. A commentary on the file format follows.

```
  Diflucan (fluconazole)
WJ BOSS4.2          3D

 34 36  0  0  1  0            999 V2000
    1.0000    1.0000    0.0000 C   0  0  0  0  0  0
    1.6558    1.4713    1.3367 C   0  0  0  0  0  0
    0.3957   -0.3571    0.0000 N   0  0  0  0  0  0
    1.7140    1.0274   -0.8243 H   0  0  0  0  0  0
    0.2102    1.6969   -0.2809 H   0  0  0  0  0  0
    2.0649    2.9638    1.2364 C   0  0  0  0  0  0
    0.8113    1.2715    2.6136 C   0  0  0  0  0  0
    2.8761    0.7590    1.4548 O   0  0  0  0  0  0
    2.9299    3.2485    0.0642 N   0  0  0  0  0  0
    2.5953    3.2856    2.1346 H   0  0  0  0  0  0
    1.1702    3.5863    1.1852 H   0  0  0  0  0  0
    2.6389    4.0760   -0.9518 C   0  0  0  0  0  0
    4.1456    2.6586   -0.1451 N   0  0  0  0  0  0
    3.6559    4.0183   -1.7909 N   0  0  0  0  0  0
    1.7482    4.6777   -1.0746 H   0  0  0  0  0  0
    4.5671    3.1780   -1.3140 C   0  0  0  0  0  0
    5.5067    2.9437   -1.8006 H   0  0  0  0  0  0
    1.4068    1.3086    3.9040 C   0  0  0  0  0  0
   -0.5953    1.1075    2.5358 C   0  0  0  0  0  0
    0.6234    1.1467    5.0625 C   0  0  0  0  0  0
    2.7342    1.5333    4.0568 F   0  0  0  0  0  0
   -0.7638    0.9553    4.9621 C   0  0  0  0  0  0
    1.0923    1.1705    6.0343 H   0  0  0  0  0  0
   -1.3736    0.9416    3.6974 C   0  0  0  0  0  0
   -1.5115    0.7994    6.0795 F   0  0  0  0  0  0
   -2.4417    0.7958    3.6192 H   0  0  0  0  0  0
   -1.1061    1.0878    1.5846 H   0  0  0  0  0  0
    3.5305    1.2501    0.9434 H   0  0  0  0  0  0
    0.9123   -1.4674    0.5516 C   0  0  0  0  0  0
   -0.7999   -0.6477   -0.5941 N   0  0  0  0  0  0
    0.0604   -2.4494    0.3148 N   0  0  0  0  0  0
    1.8446   -1.5419    1.0958 H   0  0  0  0  0  0
```

```
   -0.9669   -1.9641   -0.3727 C   0  0  0  0  0  0
   -1.8156   -2.5467   -0.7067 H   0  0  0  0  0  0
  1  2  1  0  0  0
  1  3  1  0  0  0
  1  4  1  0  0  0
  1  5  1  0  0  0
  2  6  1  0  0  0
  2  7  1  0  0  0
  2  8  1  0  0  0
  3 29  1  0  0  0
  3 30  1  0  0  0
  6  9  1  0  0  0
  6 10  1  0  0  0
  6 11  1  0  0  0
  7 18  2  0  0  0
  7 19  1  0  0  0
  8 28  1  0  0  0
  9 12  1  0  0  0
  9 13  1  0  0  0
 12 14  2  0  0  0
 12 15  1  0  0  0
 13 16  2  0  0  0
 14 16  1  0  0  0
 16 17  1  0  0  0
 18 20  1  0  0  0
 18 21  1  0  0  0
 19 24  2  0  0  0
 19 27  1  0  0  0
 20 22  2  0  0  0
 20 23  1  0  0  0
 22 24  1  0  0  0
 22 25  1  0  0  0
 24 26  1  0  0  0
 29 31  2  0  0  0
 29 32  1  0  0  0
 30 33  2  0  0  0
 31 33  1  0  0  0
 33 34  1  0  0  0
M  END
>  <MOLNAME> (MFCD00004144)
  fluconazole

>  <MDLNUMBER> (MFCD00004144)
MFCD00004144

>  <MOLCODE> (MFCD00004144)
634
$$$$
```

As shown in the example above, the standard MDL SD mol file format is comprised by the elements below. Check to ensure that your mol files are formatted as demonstrated.

Line 1 – Molecule name (QikProp retains the first 40 characters for printing.)
Line 2 - Information on the origin of the file (ignored by QikProp)
Line 3 – Blank
Line 4 – Atom and bond counts
Atom entries (x, y, z in ångströms, atomic symbol)
Bond entries (ignored by QikProp)
Any additional entries (ignored by QikProp except as noted below)
$$$$ (The data group for each molecule in a file is followed by this delimiter.)

Some additional entries to standard mol files are processed as follows. If there is an entry called <MOLNAME>, QikProp reads the following line for the molecule name that is output in all files. If there is an entry called <MDLNUMBER> or <MOLCODE>, this entry is output after the molecule name in the *<jobname>.propout* file.

D.2 Sample Maestro-formatted File

Below is the structural information for Diflucan—the same molecule whose mol file was shown in section D.1 "Sample mol File" on page 93. The difference is that this is a Maestro-formatted structure file.

```
{
  s_m_m2io_version
  :::
  1.0.0
}

f_m_ct {
  s_m_title
  s_sd_MOLNAME
  s_sd_MDLNUMBER
  :::
  "  Diflucan (fluconazole)"
  "  fluconazole  "
  MFCD00004144
  m_atom[34] {
    # First column is atom index #
    i_m_mmod_type
```

```
    r_m_x_coord
    r_m_y_coord
    r_m_z_coord
    i_m_residue_number
    s_m_insertion_code
    s_m_mmod_res
    s_m_chain_name
    i_m_color
    r_m_charge1
    r_m_charge2
    s_m_pdb_residue_name
    s_m_pdb_atom_name
    s_m_grow_name
    i_m_atomic_number
    i_m_formal_charge
    s_m_atom_name
    :::
1  3   1.000000  1.000000  0.000000 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
2  3   1.655800  1.471300  1.336700 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
3  25  0.395700 -0.357100  0.000000 0 " " X " " 4  0.00000 0.00000 "UNK" " " " " 7 0 " "
4  41  1.714000  1.027400 -0.824300 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
5  41  0.210200  1.696900 -0.280900 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
6  3   2.064900  2.963800  1.236400 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
7  2   0.811300  1.271500  2.613600 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
8  16  2.876100  0.759000  1.454800 0 " " X " " 16 0.00000 0.00000 "UNK" " " " " 8 0 " "
9  25  2.929900  3.248500  0.064200 0 " " X " " 4  0.00000 0.00000 "UNK" " " " " 7 0 " "
10 41  2.595300  3.285600  2.134600 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
11 41  1.170200  3.586300  1.185200 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
12 2   2.638900  4.076000 -0.951800 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
13 25  4.145600  2.658600 -0.145100 0 " " X " " 4  0.00000 0.00000 "UNK" " " " " 7 0 " "
14 25  3.655900  4.018300 -1.790900 0 " " X " " 4  0.00000 0.00000 "UNK" " " " " 7 0 " "
15 41  1.748200  4.677700 -1.074600 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
16 2   4.567100  3.178000 -1.314000 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
17 41  5.506700  2.943700 -1.800600 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
18 2   1.406800  1.308600  3.904000 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
19 2  -0.595300  1.107500  2.535800 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
20 2   0.623400  1.146700  5.062500 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
21 56  2.734200  1.533300  4.056800 0 " " X " " 8  0.00000 0.00000 "UNK" " " " " 9 0 " "
22 2  -0.763800  0.955300  4.962100 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
23 41  1.092300  1.170500  6.034300 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
24 2  -1.373600  0.941600  3.697400 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
25 56 -1.511500  0.799400  6.079500 0 " " X " " 8  0.00000 0.00000 "UNK" " " " " 9 0 " "
26 41 -2.441700  0.795800  3.619200 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
27 41 -1.106100  1.087800  1.584600 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
28 42  3.530500  1.250100  0.943400 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
29 2   0.912300 -1.467400  0.551600 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
30 25 -0.799900 -0.647700 -0.594100 0 " " X " " 4  0.00000 0.00000 "UNK" " " " " 7 0 " "
31 25  0.060400 -2.449400  0.314800 0 " " X " " 4  0.00000 0.00000 "UNK" " " " " 7 0 " "
32 41  1.844600 -1.541900  1.095800 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 " "
33 2  -0.966900 -1.964100 -0.372700 0 " " X " " 2  0.00000 0.00000 "UNK" " " " " 6 0 " "
34 41 -1.815600 -2.546700 -0.706700 0 " " X " " 21 0.00000 0.00000 "UNK" " " " " 1 0 "
"
    :::
  }
  m_bond[72] {
    # First column is bond index #
    i_m_from
```

```
i_m_to
i_m_order
:::
1           1       2    1
2           1       3    1
3           1       4    1
4           1       5    1
5           2       1    1
6           2       6    1
7           2       7    1
8           2       8    1
9           3       1    1
10          3      29    1
11          3      30    1
12          4       1    1
13          5       1    1
14          6       2    1
15          6       9    1
16          6      10    1
17          6      11    1
18          7       2    1
19          7      18    2
20          7      19    1
21          8       2    1
22          8      28    1
23          9       6    1
24          9      12    1
25          9      13    1
26         10       6    1
27         11       6    1
28         12       9    1
29         12      14    2
30         12      15    1
31         13       9    1
32         13      16    2
33         14      12    2
34         14      16    1
35         15      12    1
36         16      13    2
37         16      14    1
38         16      17    1
39         17      16    1
40         18       7    2
41         18      20    1
42         18      21    1
43         19       7    1
44         19      24    2
```

```
    45          19      27    1
    46          20      18    1
    47          20      22    2
    48          20      23    1
    49          21      18    1
    50          22      20    2
    51          22      24    1
    52          22      25    1
    53          23      20    1
    54          24      19    2
    55          24      22    1
    56          24      26    1
    57          25      22    1
    58          26      24    1
    59          27      19    1
    60          28       8    1
    61          29       3    1
    62          29      31    2
    63          29      32    1
    64          30       3    1
    65          30      33    2
    66          31      29    2
    67          31      33    1
    68          32      29    1
    69          33      30    2
    70          33      31    1
    71          33      34    1
    72          34      33    1
    :::
  }
}
```

The name of each field is listed near the top of the file, then the data is included in one row for each atom. For example, for the first atom listed, the first number is the atom index, the second is the atom type, the next three give the cartesian coordinates, and so on. Bond information follows the atom information in a similar format.

Index

Symbols

#acid 4
#amide 4
#amine 4
#metabol 5
#nitro 4
#rctvFG 4
#rotor 4
#stars 4
.CSV file 48
.mae file 47
.out file 48
.qpsa file 48
.warning file 49

A

accptHB 5
ACxDN^.5/SA 5
AffyPCaco 5
AffyPMDCK 5
analyses, trend vector 6
atoms, picking 10
Auto-Help 67

B

Balloon Help 67
BIPCaco 5

C

CNS 4
Command Input Area 7
concatenated mol files 52
conventions, document 1

D

dip^2/V 5
dipole 4
DISPLAY, setting 16
donorHB 5

E

EA(eV) 5
entries 8
  excluding 9
  including 9
environment variables, setting 15

F

fast processing mode 26
file i/o directory 49
FISA 4
FOSA 4

G

glob 5

H

host selection 41, 42
Hydrogen/Lone-Pair Fixup panel ... 24

I

IP(ev) 5

J

jobcontrol utility
  purging database with 76
  syntax for 75
jobs
  record 78
  remote submission 70, 77
  scratch directory for 69, 77

M

Maestro
  Main Application Window 7
  workflow paradigm 10

working directory, specifying...20
Main Application Window.....7
molecule name.....4
Monitor panel 25
mouse functions 11
MW 4

N

New Series dialog 34

O

online help.....67
-out.mae file 47

P

Pick State menu 10
pick states.....10
picking 10
picking mode 11
PISA 4
Plot XY panel 35
Preferences panel 49
project entries, sorting 27
Project Facility, introduction to 8
Project Table panel.....23
projects 8
properties and descriptors
#acid 4
#amide 4
#amine 4
#metabol 5
#nitro 4
#rctvFG.....4
#rotor 4
#stars.....4
accptHB 5
ACxDN^.5/SA.....5
AffyPCaco.....5
AffyPMDCK 5
BIPCaco.....5
CNS 4
dip^2/V 5
dipole.....4
donorHB 5
EA(eV) 5
FISA 4
FOSA.....4
glob.....5
IP(ev).....5
molecule name.....4
MW 4
PISA 4
properties 4
QPlogBB.....5
QPlogKhsa 5
QPlogKp 5
QPlogPC16.....5
QPlogPo/w 5
QPlogPoct 5
QPlogPw 5
QPlogS.....5
QPpolrz.....5
SASA.....4
WPSA.....4
properties, plotting 34
properties, predicting.....33

Q

QikFit
.myfits file 59
.out file.....59
launching.....17
running.....31, 56
significance cutoff values 58
test file 58
training file 57
qikfit command
arguments.....56
help for.....68
-HOST option 59
syntax.....56

-WAIT option 57
QikProp
command line..... 16
launching from Maestro 16
qikprop command
help for 68
input file types..... 52
qikprop command syntax 51
QikProp jobs
incorporation of output 44
input for..... 43
monitoring 45
multiple molecule processing .. 52
naming..... 41
output files
.CSV 48
.mae 47
.out..... 48
.qpsa..... 48
.warning..... 49
-out.mae..... 47
starting..... 44
QikProp panel 24
accessing 41
QikSim
input 62
launching 17
output 38
running 38
qiksim command
arguments 61
help for 68
syntax 61
QPlimits file..... 3, 26, 43
QPlogBB 5
QPlogKhsa..... 5
QPlogKp..... 5
QPlogPC16 5
QPlogPo/w..... 5
QPlogPoct..... 5
QPlogPw 5
QPlogS 5
QPmyfits file..... 33
QPpolrz 5

R

restrictions
for input structures..... 52
row..... 8

S

SASA 4
scatter plot, creating..... 36
SCHRODINGER, setting..... 15
schrodinger.hosts file 71
Sidebar 13
Sort Project Table panel 27
structures
importing 22
restrictions 52
rotating 11
translating 11

T

technical support 68
test file, generating 30
training file
creating..... 29
excluding data columns from... 56
new property data in 56
training set, generating..... 24
Transformation bar 13
trend vector analyses..... 6
tutorial directory, creating 21

V

volume..... 4

W

workflow paradigm 10
workflow, project based..... 8

working directory, specifying.....20
WPSA.....4

Z

zooming 11

1500 SW First Avenue
Suite 1180
Portland, OR 97201

32nd Floor, Tower 45
120 West Forty-Fifth Street
New York, NY 10036

3655 Nobel Drive
Suite 550
San Diego, CA 92122

SCHRÖDINGER